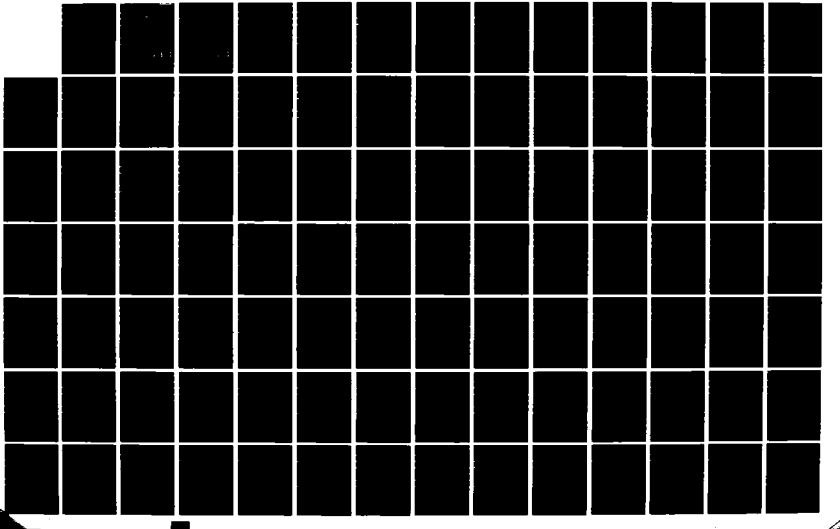
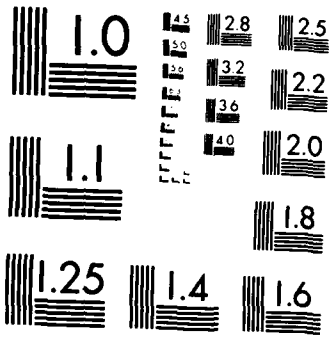
AD-A151 699 THE FIRST CORTICAL IMPLANT OF A SEMICONDUCTOR MULTIELECTRODE ARRAY: CONTI..(U) AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOOL OF ENGI.. 1/3

UNCLASSIFIED J D COLLIER ET AL. DEC 84 F/G 5/10 NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963 A

THE FIRST CORTICAL IMPLANT OF A SEMICONDUCTOR MULTIELECTRODE ARRAY: CONTINUED DATA ANALYSIS

THESIS

John D. Collier
Captain, USAF

Deanne B. Mahoney
Captain, USAF

AFIT/GE/ENG/84D-39

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY
AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/GE/GSO/ENG/84D-20

THE FIRST CORTICAL IMPLANT OF A SEMICONDUCTOR MULTIELECTRODE ARRAY: CONTINUED DATA ANALYSIS

THESIS

John D. Collier
Captain, USAF

Deanne B. Maloney
Captain, USAF

AFIT/GE/GSO/ENG/84D-20

Approved for public release; distribution unlimited

AFIT/GE/GSO/ENG/84D-20

THE FIRST CORTICAL IMPLANT OF A SEMICONDUCTOR MULTIELECTRODE ARRAY: CONTINUED DATA ANALYSIS

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
In Partial Fulfillment of the
Requirements for the Degree of
Master of Science in Electrical Engineering
and the Degree of
Master of Science in Operations Research

John D. Corrier, B.S.
Captain, USAF

Deanne B. Mahoney, B.S.
Captain, USAF

December 1984

Preface

The purpose of our study was to complete the analysis of brain data from the first cortical implant of a semiconductor multielectrode array. We hope that our efforts here will aid further tests of multielectrode array implants. Although we were unable to complete all our desired objectives because of various problems and delays, we feel that we analyzed most of the data and hopefully answered some questions raised by the previous analysis efforts.

We wish to thank Dr. Matthew Kabrisky and Lt. Col. Joseph Coleman for their help in completing this thesis. They gave us guidance and encouragement whenever we needed it the most.

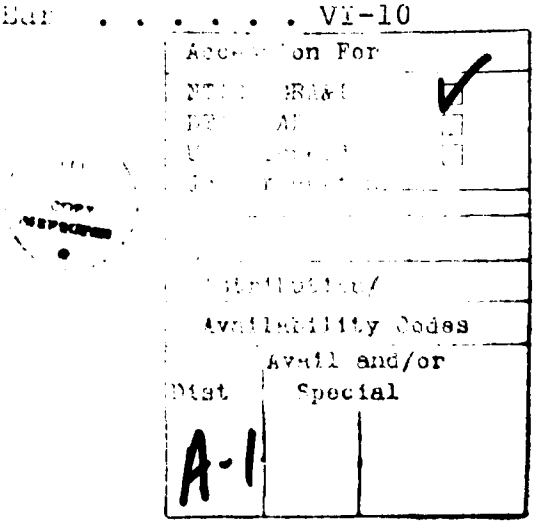
We also wish to thank Capt. Douglas Fitzpatrick for assisting us in overcoming the learning curve of the MASSCOMP computer and for also being a tremendous help in designing the computer programs used in analyzing the data.

Also, we appreciate the assistance of Mr. Paul Fisher of AFWAL. He loaned us the MASSCOMP Digital-to-Analog Converter module which enabled us to extract the analog EEG data.

Special thanks goes to our families, whose patience and understanding gave us the fortitude to complete this thesis.

Table of Contents

List of Figures



List of Tables

Abstract

Previous research conducted at the Air Force Institute of Technology (AFIT), collected and partially analyzed data from a multielectrode array implanted on the visual cortex of a dog's brain. The work presented here is the completion of this data analysis. Electroencephalogram (EEG) recordings were made and Visual Evoked Response (VER) techniques were used on most of the data. The EEG strip chart recordings display evidence of low frequency alpha waves, thus indicating that the semiconductor array implant is a suitable method for extracting brain data. Summation/averaging of the data from individual electrodes illustrate that the strobe stimulus leaked into the data collection circuitry. There is evidence that the dog probably became accustomed to the strobe signals and also that the AFIT array showed signs of degradation while implanted on the cortex. A possible evoked response was found during the latter days of testing, but because of the suspected deterioration of the AFIT array, this cannot be held as certain. Included are recommendations on how the next family of AFIT multielectrode array IC's data collection and analysis should be conducted.

THE FIRST CORTICAL IMPLANT OF A SEMICONDUCTOR MULTIELECTRODE ARRAY: CONTINUED DATA ANALYSIS

I. Introduction

Background

The mystery of how the brain functions has been the subject of speculation for centuries. In recent decades however, scientists and engineers have been investigating specifically how the cerebral cortex, present only in the mammalian brain, processes functions such as; hearing, speech, vision, thought, etc. Research in the above areas seems to have intensified in the last thirty years, since the advent of the transistor made possible the development of more accurate and stable instrumentation, to measure and record cerebral cortex electrical activity.

Now, with the recent progress in very large scale integration (VLSI) ICs, not only is the accurate measurement of cerebral cortex electrical activity possible, but also the simulation of cerebral neuron activity seems to be a very realistic goal. Of particular interest to researchers in the field of pattern recognition is the operation of the human visual system. Unlocking the secrets of the human visual system may lead to breakthroughs in the area of visual scene analysis (23:3). Visual scene analysis covers such diverse subjects as target acquisition, machine inspection of manufactured parts, reading machines, and electroencephalogram

(EEG) analysis; to name but a few.

The most common methods of measuring human visual system activity in the cerebral cortex are the electroencephalogram and the visual evoked response (VER). These measurement techniques were initially developed using scalp electrodes on the subject, normally human, and analyzed in an attempt to discover something about brain function (18:I-1). Visual evoked response refers to the average or summation of cortical activity recorded from the scalp electrodes while the eye is exposed to a known input signal of repeated strobe flashes or geometric patterns. Although these signals are useful, they can influenced by such things as drugs, diseases, anesthesia level, and attention. They also represent summation of millions of local signals and as a result, attempts to model the visual recognition function using EEG or VER seem to be analogous to analyzing the internal workings of an inconsistent machine by listening to the noise it makes (19:1).

Since external measurements did not seem to be able to provide the fine grain data needed to model the visual system, new procedures based on electrodes implanted in the brain or on its surface were developed. Early efforts attempted only gross measurements because to really model completely the total activity of the brain, the signals of millions of neurons would have to be measured and analyzed, simultaneously. Such an enormous multiple electrode, with element spacings on the order of approximately 50 microns (23:40) have not been feasible. Dewett made the largest multiple electrode known to

us before the work repeated here. He was able to achieve a 1.5 millimeter (between-electrode) spacing using 400 (20 x 20) electrodes (8,9,10). The original work by Lorente de No' (26:2) illustrated that the anatomy of the cerebral cortex was remarkably uniform. This uniformity suggests that the processing performed by different functional areas of the cerebral cortex is dependent on the input signals (18:1-2). If this is in fact the case, then the examination of the basic functional elements in a single area of the cortex could provide answers which may be universally applicable to other brain functions.

An indication that these basic functional elements could be columnar arrangements of neurons is suggested by several researchers (18:1-3). After considering the neuron pattern or "wiring diagram" of the eye brain system (26), Kabrisky, and Hubel and Wiesel, among others, suggest that these columns may be the best level at which to obtain the vision system model (20,23). The columnar arrangement of neurons are labeled basic computational elements (BCE's) and are thought to have at least one input channel and several output channels at least in the primary visual center. The BCE's seem to be interconnected to different areas in the cortex; for example, interconnections have been mapped from the primary to the secondary visual cortex areas (18:1-3).

Measurements of eye-brain activity at the BCE level appears to be the one with the highest probability of producing a model of the vision system which. To facilitate the measure-

levels (i.e., the clock had to operate between 0 and +5 volts, where +5 represented logic state "1" and had to change state within a few hundred nanoseconds.) (18:II-6)

The Time Delay Detector. The external clock generator converted the synch into a TTL compatible square wave, but since this would only pick up one of the four row signals, a time delay was needed to allow positioning the pulses of the external clock within each sampling window. The resulting TTL signal (the clock) had a maximum time delay of about 3.5 msec (18:II-6).

The External Clock Controller. The external clock controller was designed to start the clock at any point in the data and stop it after a designated number of samples had been taken. This was accomplished by the circuit counting the number of clock pulses and then turning the external clock on or off when a start or stop count, set through the data entry controller, was reached. This ensured the clock would always start at the desired point in the data (18:II-6).

The external clock controller centered around two eight-digit presetable counters. The data entry controller allowed up to eight digits to be entered and stored in the memory of the desired counter. Counter one was used to count the number of synch pulses, so its memory was preset to the designed starting point for the clock. Counter two counted the number of clock pulses transmitted to the A-D converter, so its memory was preset to the desired number of samples (18:II-6).

was available in the AFIT Signal Processing Laboratory. Since the data were recorded in analog form, they had to be digitized and demultiplexed.

Hayes' thesis was divided into three areas:

1. The synchronization (sync) signal conditioning phase.
2. Digitization and pre-analysis phase.
3. Data analysis phase.

The synchronization signal conditioning phase was the most complex area of Hayes' thesis effort. He was required to design and build five circuits to accomplish the pulse shaping and timing necessary to digitize the data. The circuits built were an external clock generator, an external clock controller, a data entry controller, a mode controller, and a time delay detector.

Circuits Built.

The External Clock Generator. To make sure the Analog-to-Digital Converter of the Eclipse was synchronized with the data, the external clock option was chosen on the computer. Using this, the digitizing and demultiplexing could be done simultaneously. The synch signal on the tape was used as the external clock source. Because of limitations of the Eclipse computer, and the actual form of the synch signal (It was not compatible with standard transistor-transistor logic or TTL), the synch could not be used directly. A circuit was designed which would sharpen and convert the signal to TTL

Table 1

Recorder Channels

SIGNAL DEFINITION	RECORDER CHANNEL
1. External Sync (Row One Tape On) 1 Hz	#3
2. Strobe (250 Hz)	#4
3. Voice Track	#5
4. Array Data Amplifiers	
Amplifier 1	#8
Amplifier 2	#10
Amplifier 3	#12
Amplifier 4	#14
Amplifier 5 - Differential External Electrode On Dog's Ear WRT The Indifferent Electrode On The Array	#11
5. 1 KHz Clock	#7

1 Khz source signal (not collected during the first few days of testing).

Hayes

Hayes was the first extensive effort to analyze the data collected by Hensley and Denton. The computer chosen to do the work was a Data General Eclipse S/250 minicomputer, which

Data Collection. Two types of data were collected; physiological data and cortical bioelectrical data. The Hensley and Denton thesis is only concerned with the cortical bioelectrical data. The data were collected over a period of seventeen days. Data were collected every day for the first twelve days, and then every other day. For the first several days, data were collected in the differential mode only (see explanation in chapter I). For the remainder of the days, collections were done in both differential and absolute modes.

The Strobe. In order to extract response information from the visual cortex a stimulus must be used. In this case a strobe lamp was used to provide the stimulus to produce VER data. Two types of strobes were used during the testing. The first strobe, a small incadescent lamp was used for the first six days. It was turned on for an average of 180 milliseconds (ms), during the pulse period of 500 ms. The frequency or this strobe is 2 Hz. The second strobe using a flash tube was only on for 5 ms. The "on-time" was shortened to reduce any corrupting effects the square wave strobe may have imposed on the decay period of the VER. The second strobe had a frequency or 1 Hz, the period being 1000 ms.

Storage. The data were stored on an Ampex FR 1300, 16 track FM recorder (14 data tracks and two voice tracks). Table 1 shows what was stored on each channel. Besides the four channels with the output raw data and the voice track channel, there is the raw one (A1) input signal or synch and a

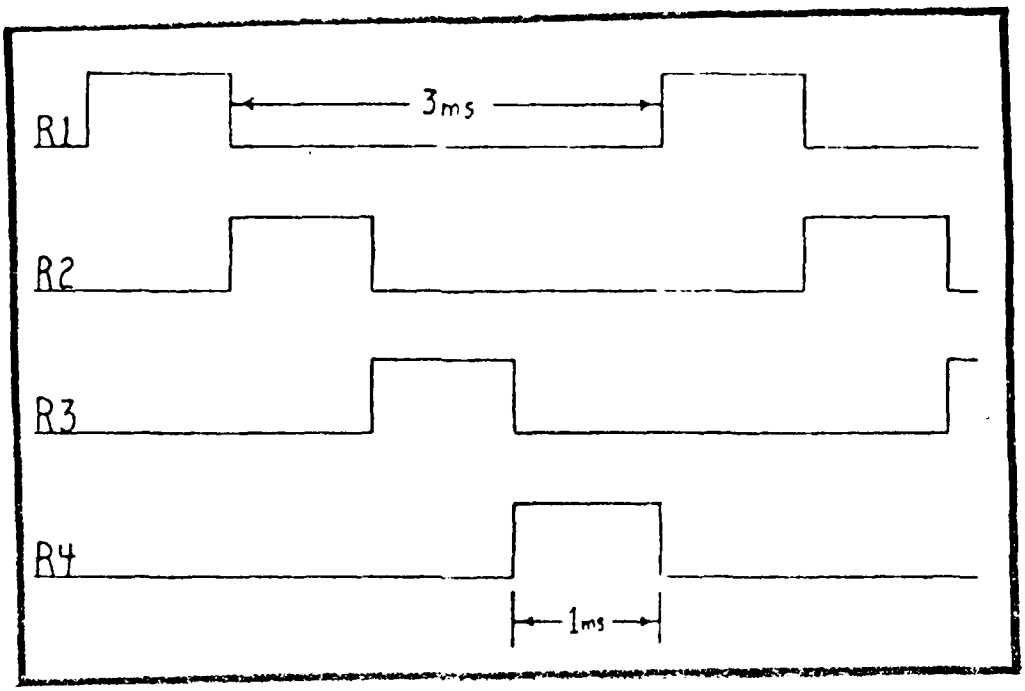


Figure 2-2. Row Input (19:30)

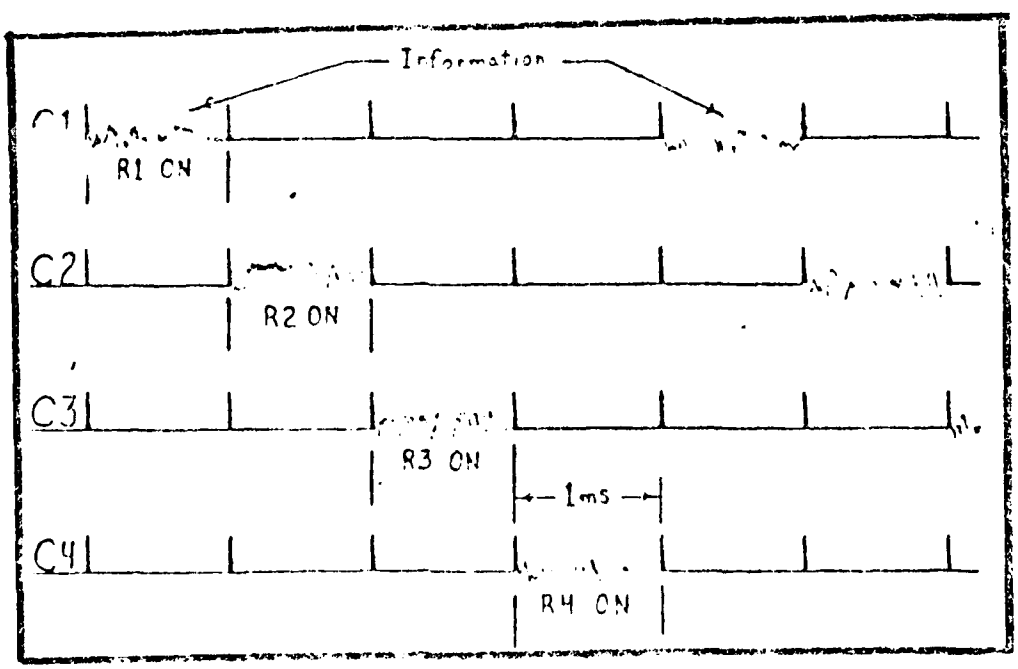


Figure 2-3. Column Output (19:30)

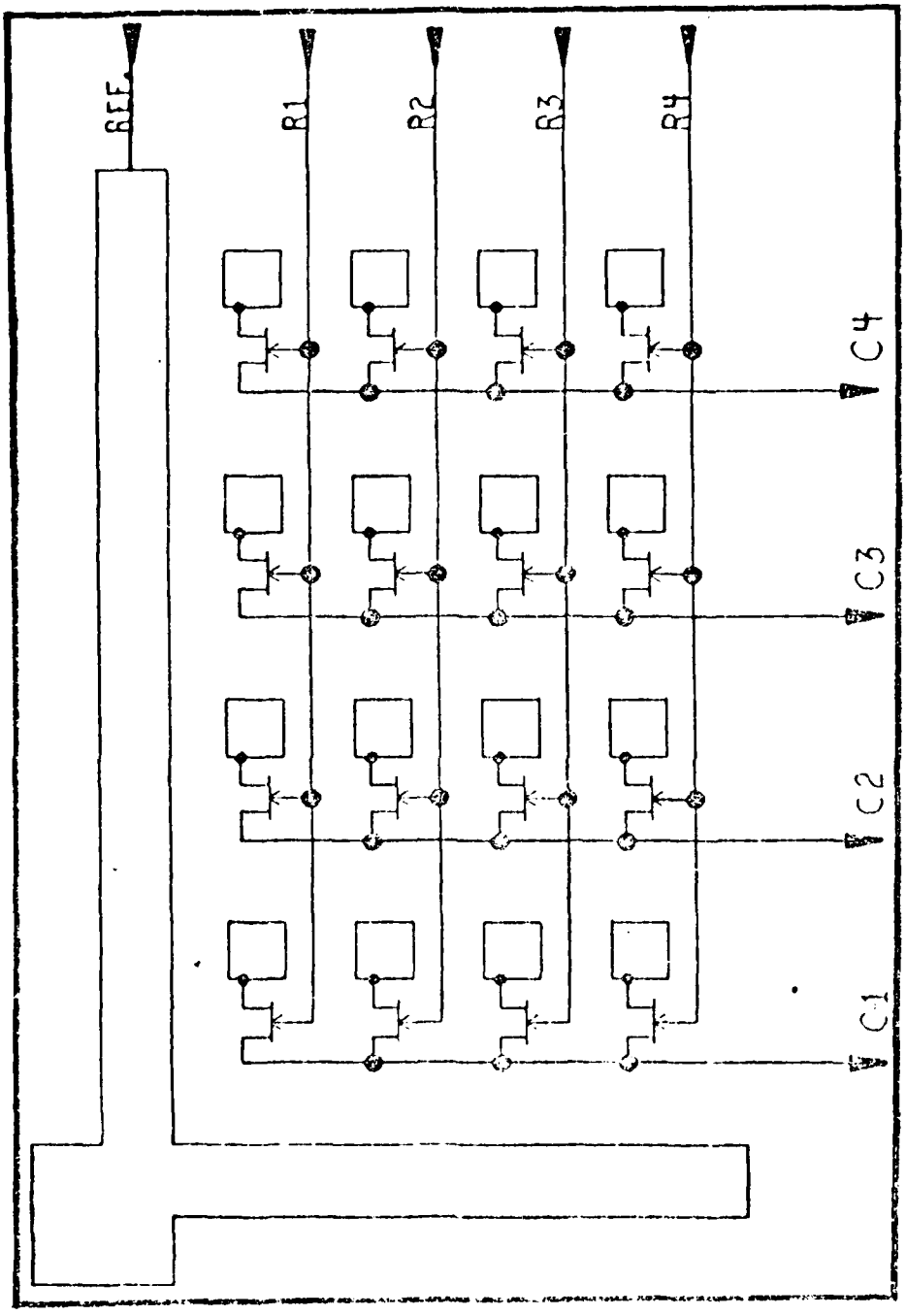


Figure 2-1

Hensley and Denton's Multielectrode Array Diplane (19:25)

II. Data Acquisition and Analysis History

The principle background for this thesis is included in the work of Hensley and Denton and that of Hayes. Hensley and Denton designed the semiconductor implant and collected the data used here; Hayes conducted the first systematic analysis of the data. A basic understanding of these two thesis efforts is needed to understand the problems which were confronted in this thesis. This section will describe the relevant portions of the two previous theses and briefly explain the theory behind some of the actions taken here.

Hensley and Denton

The Array. The AFIT multielectrode array used was a four-by-four array of JFET's and associated individual electrodes as shown in figure 2-1. In the electrical diagram, each source is shown connected to a square "sensing pad" (i.e. electrode) through which contact is made with the visual cortex. The series connection of the JFET gates and drains make up the array "rows" and "columns", respectively. A reference electrode (REF) is also provided (19:28).

When multiplexed signals (Figure 2-2) are applied to the rows, information present on the sources of a particular row are passed through the JFET drains to the column outputs. Thus, sequential selection of the rows produces a multiplexed output at each column. Figure 2-3 shows the basic format of the column output (19:28).

coincide with the three main areas addressed in the problem statement.

Chapter 2 will present a summary of the development of the AFIT multielectrode array, and the theory which this thesis will use. This is to include a summary of Hensley and Denton's thesis and also Hayes' thesis.

Chapter 3 discusses the hardware necessary to condition the timing signals, so that a TTL signal will be presented to the A/D conversion device. Additionally, the hardware development necessary to synchronize the timing signals with the data signals is presented.

Chapter 4 addresses, initially, the familiarization with the real-time data acquisition processor, the MASSCOMP MC-500. Next, the software design necessary to perform the A/D conversion, the VER, and the video movie will be discussed.

Chapter 5 presents the data reduction details. Specifically, how the data were transferred from the multichannel tape recorder and how the VER data were obtained.

Chapter 6 discusses the results and analysis drawn on the data collected, as to its validity and to whether or not additional information was obtained to help model the eye-brain system.

Chapter 7 presents conclusions and recommendations on data collection and real-time data analysis using the MASSCOMP MC-500 Multichannel Real-Time Processor with future multielectrode array implants.

Scope

Data collected by Hensley and Denton (19) were recorded on 14 of the 17 days during which the array was implanted. The number of tests ranged from two to four depending on whether or not the dog was or was not anesthesized. Each test was conducted under slightly different conditions, and care must be taken to analyze the results accordingly. First, a A/D conversion must be done and all the data demultiplexed. Next, the software to compute both the VER and the other signal processing algorithms mentioned above, must be developed. This software may have to be tailored to accommodate the parameters peculiar to the various individual tests.

Assumptions

The data were originally collected at a clock rate of 250 Hertz (Hz) because most information published on the frequency of brain signals indicates that 25 Hz is the maximum frequency component (19:33). If this assumption is true, the Nyquist Criteria (sampling of a signal must be greater than twice the maximum frequency) has been satisfied. It has; therefore, been decided to retain 250 Hz as the data sampling rate for the A/D conversion.

Approach and Presentation

The objective of this research is the completion of the data analysis begun by both Hensley and Denton (19) and Hayes (18). The presentation will essentially be ordered to

[illegible] must be overcome before the processor can be used effectively. Although, this special purpose processor will simplify the timing problems faced by Hayes (18), care must still be taken to ensure that the data and the strobe signal are in sync. In this way, a valid averaged evoked response can be computed.

The second part of the problem is the analysis of the data using signal processing techniques. The first analysis method to be applied is the averaged evoked response which extracts the evoked response from the background EEG data. Software will be developed to perform the averaging of the synchronized data files. Once this is accomplished additional signal processing techniques will be applied, time permitting. Some of the most common methods used to distinguish the signal from the noise are Kalman filtering (21), Fourier and Walsh transformations (22,41), Weiner and time-varying filtering (11), and amplitude histograms (4).

Finally, a video tape will be made of the changing signal levels at each of the 16 array electrodes. This analysis may give information on the firing of the BCE's as the subject was stimulated with the strobe light. Again, this will be primarily a software effort to transport the data files from the special purpose computer to the Eclipse processor in the signal processing laboratory, building 640.

A secondary objective of this thesis is to develop algorithms and procedures which can be used by the next larger array implantation experiment.

Because of time limitations caused by circuit construction and processor limitations, however, the data analysis was not completed by Hayes.

Statement of Problem

Currently, work is underway to design and fabricate a larger array with 256 electrodes. However, before the second implant operation commences, the data from the first implant should be analyzed in order that a decision, as to the validity of the data, can be made. Therefore, the main effort of this thesis will be to complete the data analysis process.

There are three main areas which will be investigated in an effort to solve this problem. First, an analog-to-digital (A/D) conversion must be performed on the data. This step discretizes the data signals which allows the application of signal processing algorithms. A special purpose real-time data acqusition processor has been recently acquired by the AFIT Signal Processing Laboratory which can perform an A/D conversion on up to 16 channels of data, simultaneously. In this way, the synchronization of the data with the timing signals can be accomplished without extensive and complex external timing circuitry. Thus, software must be written to perform the A/D conversion and demultiplex the column and row data of each array electrode into separate files. However, because this computer is so new a defi-

nitrogen environment) (19:5). Finally, with the assistance of personnel from the Aerospace Medical Research Laboratories, Wright-Patterson AFB, OH; the multielectrode array was implanted in the visual cortex area of a laboratory beagle. Data were collected in two modes (19):

1. The absolute mode, where the signals from each of the 16 electrodes in the array were measured with respect to a common electrode in the array.
2. The differential mode, where each electrode signal was measured with respect to the signal of the electrode in the neighboring column.

Hensley and Denton were unable to perform a detailed analysis of the data because of time limitations. Therefore, additional thesis work was performed by Rodney E. Hayes (18) to analyze and determine the validity of the data collected from the implanted array. In his research effort Hayes needed to solve the following three problems:

1. Circuitry was required to condition the data timing signals because the processor used to perform the analog-to-digital conversion on the data signals operated with transistor-transistor logic (TTL). This means that the clock must operate between 0 and +5 volts.
2. A control circuit was needed to ensure the synchronization of the strobe data signal and the data from the array electrodes to compute the average evoked response (AER).
3. Finally, once the data were digitized, software had to be developed to permit analysis of the data.

succeeded in fabricating a device which operated in air, the device failed when immersed in a saline solution similiar to that found on the surface of the cortex. Thus, further research into a method of encapsulating the switching circuitry was required to protect the chip against sodium ion contamination.

The next thesis concerning the multielectrode array was written by George German (16), and specifically addressed the area which caused the device fabricated by Fitzgerald to fail, that of circuit passivation. German evaluated sixteen different combinations of materials and application processes; however, only two materials were found which would resist the degrading effects of CSF, phosphosilicate glass (PSG) and polymide (18:36). German designed and tested the drive circuitry which provided the multiplexed drive signals to the array while it was immersed in a simulated CSF solution. Additionally, he noted that the Junction Field Effect Transistors (JFET) circuitry exhibited non-uniform switching characteristics.

The implant of the semiconductor multielectrode array into the visual cortex of a dog was engineered by Russell W. Hensley and David C. Denton (19). They encapsulated the array with polyimide, because of the availability of the technology on Wright-Patterson AFB, to ensure the survivability of the array in the CSF environment. The external drive circuitry was redesigned to permit simulated testing without connecting the array because it required special handling (storage in a

ment of BCE interaction, several researchers have attempted to use microscopic electrodes implanted on the surface of the cortex at distances which approximate the separation of the BCEs. Some of the most significant research in the area of implanted multiple electrodes has taken place at the Air Force Institute of Technology (AFIT). There have been five previous Master's Theses which dealt with the subject of multiple electrode placement on the surface of the cerebral cortex.

The initial proposal of an AFIT multielectrode array was made by Joseph Tatman (42). He theorized that it would be possible to construct an electrode array of 100 rows by 100 columns with electrodes on 50 micron centers and have a system data rate of one million words per second. However, to test the theory of a multiplexed electrode array and to simplify construction, a four by four electrode array was fabricated. Tatman's attempt to fabricate an operational IC was unsuccessful, but his efforts did show that further work in this area was warranted, if the problems he encountered during the fabrication phase could be overcome.

The efforts begun by Tatman were carried on by Gary Fitzgerald (14). Some of the problems encountered by Tatman were solved; specifically, the lack of ohmic contact with the metalized leads and the leakage paths in the passivation layer. The passivation layer is a layer of material which protects the switching circuitry from the cerebrospinal fluid (CSF), which is primarily a saline solution, but allows the electrode surfaces to contact the cortex. Although Fitzgerald

The Mode Controller. The mode controller was used to set/reset flags within other circuits. These flags enabled or disabled the external clock generator, set the external clock controller to respond to the first or last occurrence of the stimulus control signal (the strobe), and set the external clock controller to either start or stop the external clock depending on whether normal cortical activity (strobe not present) or VER (strobe present) data were being sampled.

Digitization and Pre-analysis. Next, Hayes performed the digitization and pre-analysis phase by designing routines to:

1. Control the analog to digital to analog (A/D/A) device from the Eclipse S/250 minicomputer.
2. Scale the digitized data to between ± 5 volts before analysis.
3. Remove physical characteristics in the data which were a function of the data collection process as follows:
    a. The gain factor inserted during data collection.
    b. Electrode biases.
    c. Artifacts caused by aging of the array or changing of the pinch-off voltages in the array drive ciruitry.

Data Analysis. Analysis of the data, the final phase of the Hayes Thesis, was primarily a software design and implementation problem. Hayes either modified or developed routines to accomplish the following (18:III-11-16):

1. Produce a video display to illustrate the spatio-temporal correlation of adjacent BCE's.

2. Calculate the average evoked response (AER) at the BCE level.
3. Determine a low frequency cut-off to allow implementation of a software low-pass filter which would be used to improve the results of the previous two routines.

The spatio-temporal correlation required two routines. The first involved the construction of a topological arrangement of the data, 16 samples (one per electrode) at a time, with the amplitude of each electrode represented by a variation in grey level. Next, a routine was designed to receive output from the first routine and then to display the 16 grey levels, one 4 x 4 sample at a time, on a monitor at a variable rate. Finally, the 4 x 4 display of grey levels was observed to see if a spatio-temporal correlation existed within the data received from the surface of the cortex.

An average evoked response (AER) analysis was performed to show the relationship of the visual stimulus (strobe) to BCE activity on the cortex surface. The routine extracts the VER from the digitized EEG data by using the digitized strobe file as a marker which delineated the beginning and end of each stimulation. Then by reading the markers, the digitized EEG data was separated into blocks which were summed and averaged. The result was a plot showing the VER increasing linearly (if consistent) while the random noise tended to sum as a Root Mean Square (R.M.S.) process.

The final analysis routine involved filtering or smoothing the data to improve the results of the topographic and VER analyses. First, a finite impulse response (FIR) filtering algorithm computed the convolution sum of the filter coefficients and the adjusted data magnitudes (i.e. data rescaled, attenuated and mean subtracted to remove electrode bias). Then a moving average was computed by using either a Bartlett (triangular), Hanning, Hamming, or Blackman window.

Hayes was only able to perform a preliminary analysis on one day's worth of data due to the delays encountered during the hardware development phase. As a result, some of his conclusions will be further investigated in this thesis effort. Specifically, whether or not the shape of the EVR is dependent on the size of the data file averaged. That is, is the VER consistent during the stimulus period. Also, duplication of the VER obtained by Hensley and Denton (19) from data collected from electrodes attached on the dog's ear and the reference electrode within the AFIT Array will be attempted. Finally, answers will be sought as to how degredation of the electrodes influenced the reliability of the data.

Theory

Digitization of the Data. The first part of the thesis encompasses the digitization of the brain data obtained by Hensley and Denton. Digitizing can be accomplished by ensuring that the Nyquist Criteria is met (See Chapter I). This Theorem states, "A continuous analogue function x(t)

which has a limited Fourier spectrum, that is a spectrum x(jw) such that 2(jw)=0 for w > Wm, is uniquely described from a knowledge of its values at uniformly spaced time instants, T units apart, where T=2x/Ws and Ws >= 2Wm" (7:2). In other words, if a function x(t) has little or no energy content for frequencies above some value fm (where Wm = 2 x fm), then samples of x(t) taken at a rate fs >= 2fm will preserve the nature of x(t) and analysis of these samples will give information about the analog process.

Extracting VER. The analysis technique used to extract the VER from the normal cortical activity is summation /averaging. The data are divided into a number of sections, each section beginning when one strobe flashed and ending just before the next strobe flash. The sections can be summed, then averaged (if needed) to emphasize the evoked response. This assumes that there is a consistent brain response to the stimulus that is synchronized with the strobe while the normal cortical activity and the noise are random and correlated with the stimulus. Therefore, if waveform f(t) consists of two components: the evoked response s(t), and noise (anything else) n(t); then

$$f(t) = s(t) + n(t)$$

The evoked response should become increasingly clear with summation since it should repeat; therefore, constructively add. The noise and the normal cortical activity should not constructively add (12:206, 35:87-90, 42:39-41).

Spatio-Temporal Correlation. Another analysis technique

attempts to find any spatio-temporal correlation in the data. This can be accomplished by constructing topologic arrangements of the data, 16 samples at a time (one from each electrode), displaying these in succession on a video monitor, and examining the output to see if any correlation can be detected by visual observation.

III. Hardware Design and Implementation

Introduction

The manner in which the data were collected dictated the actions required to pre-condition the data and timing signals for analog-to-digital (A/D) conversion and subsequent analysis. Specifically, data were collected over a period of 17 days (19:81), but only eight of those days provided data which could be analyzed. The reasons some of the data could not be analyzed are as follows:

1. There was no voice track recorded on days 1, 2, and 12.
2. The row one turn-on signal (ROTOS) was not recorded on day 10 and was distorted on day 16.
3. A strobe signal was not recorded on day 14.
4. Data were recorded every other day after day 12, so there are no data for days 13, 15, and 17.

Data from the multielectrode array were originally recorded on a 16 channel FM recorder by Hensley and Denton (19). However, some of the recorder channels were found to introduce distortion in the input signals (18:11-1). Only four channels were found to introduce no additional distortion and these were used to record the output data. A block diagram of the array and the output circuitry is shown in Figure 3-1. Three other channels which did degrade the input signals were used to record the row one turn-on signal (ROTOS), the stimulus control (strobe) signal, and a 1 kHz clock signal (starting

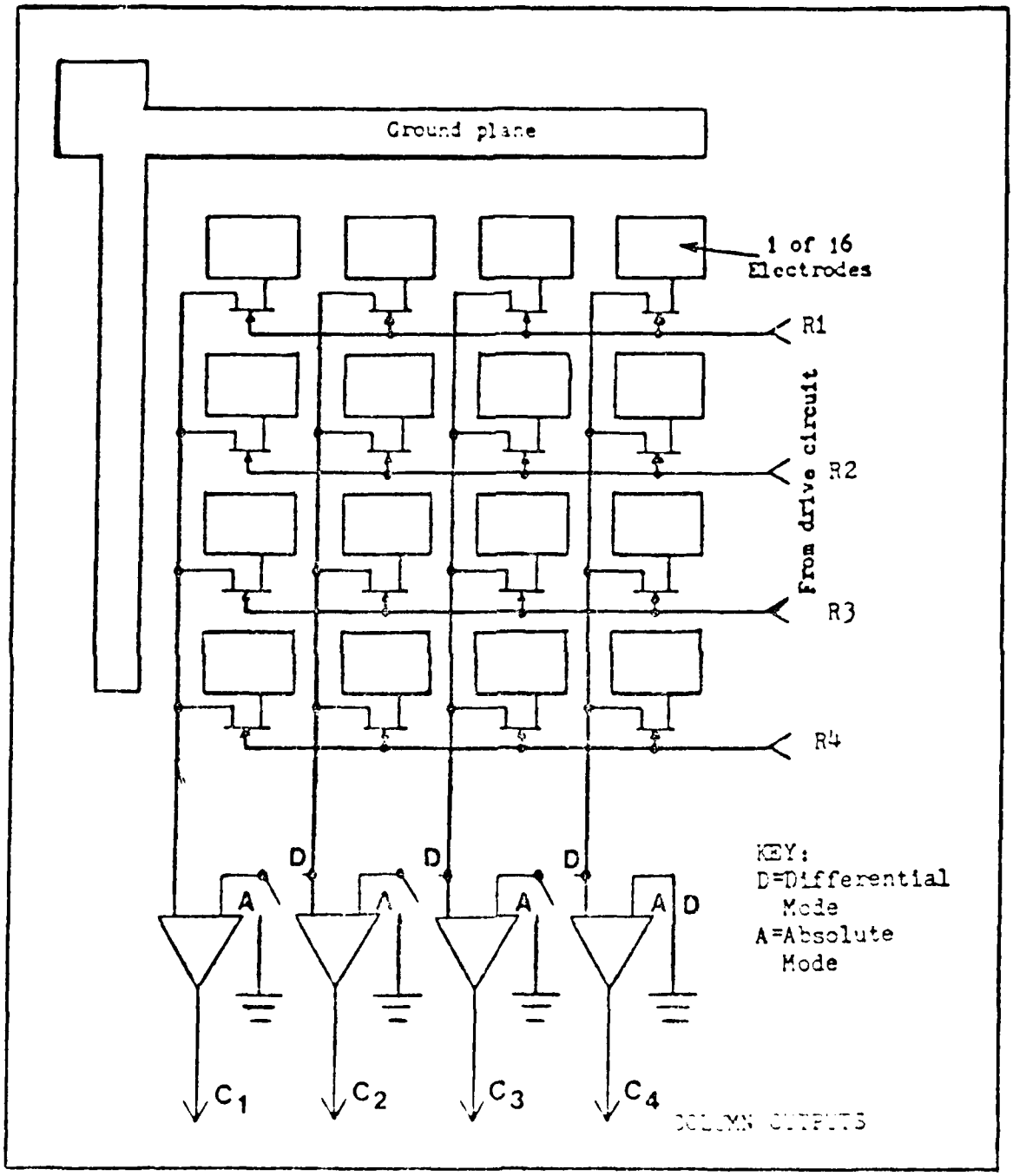


Figure 3-1

Multielectrode Array and Output Circuitry (19:28-44)

with day 4). As a result of the distortion, these digital signals would require conditioning to make a sampling pulse which would be compatible with the processor. Additionally, each signal from the array must be demultiplexed prior to digitizing the data. The ROTOS provides the means to control the timing of the demultiplexing of the data.

Adaptation of Hayes' Circuitry

Prior to digitizing the data, circuitry to shape and convert the sampling pulses to TTL levels was required. This area was addressed by Hayes (18); therefore, it was decided that a portion of the circuitry he developed could be used to accomplish the reconditioning of the sampling pulse. The circuit diagram of the External Clock Generator which is the sync or ROTOS reshaping circuit is illustrated in Figure 3-2.

Basically, the External Clock Generator first provides for ROTOS amplification using a Motorola 1748 Operational Amplifier (Op-Amp). Next, the signal was inserted into a 7414 Schmitt Trigger inverter for reshaping and conversion to a transistor-to-transistor level (TTL) square wave by:

1. Increasing the rise and fall times of the leading and trailing edges, respectively.
2. Changing the logic "1" state to +5 volts and the logic "0" state to 0 volts.

Then the square wave was input to a 74123 Dual Multivibrator, which in conjuction with a 72555 Timer IC, allowed for both pulse positioning and a variable pulse width clock. In follow-

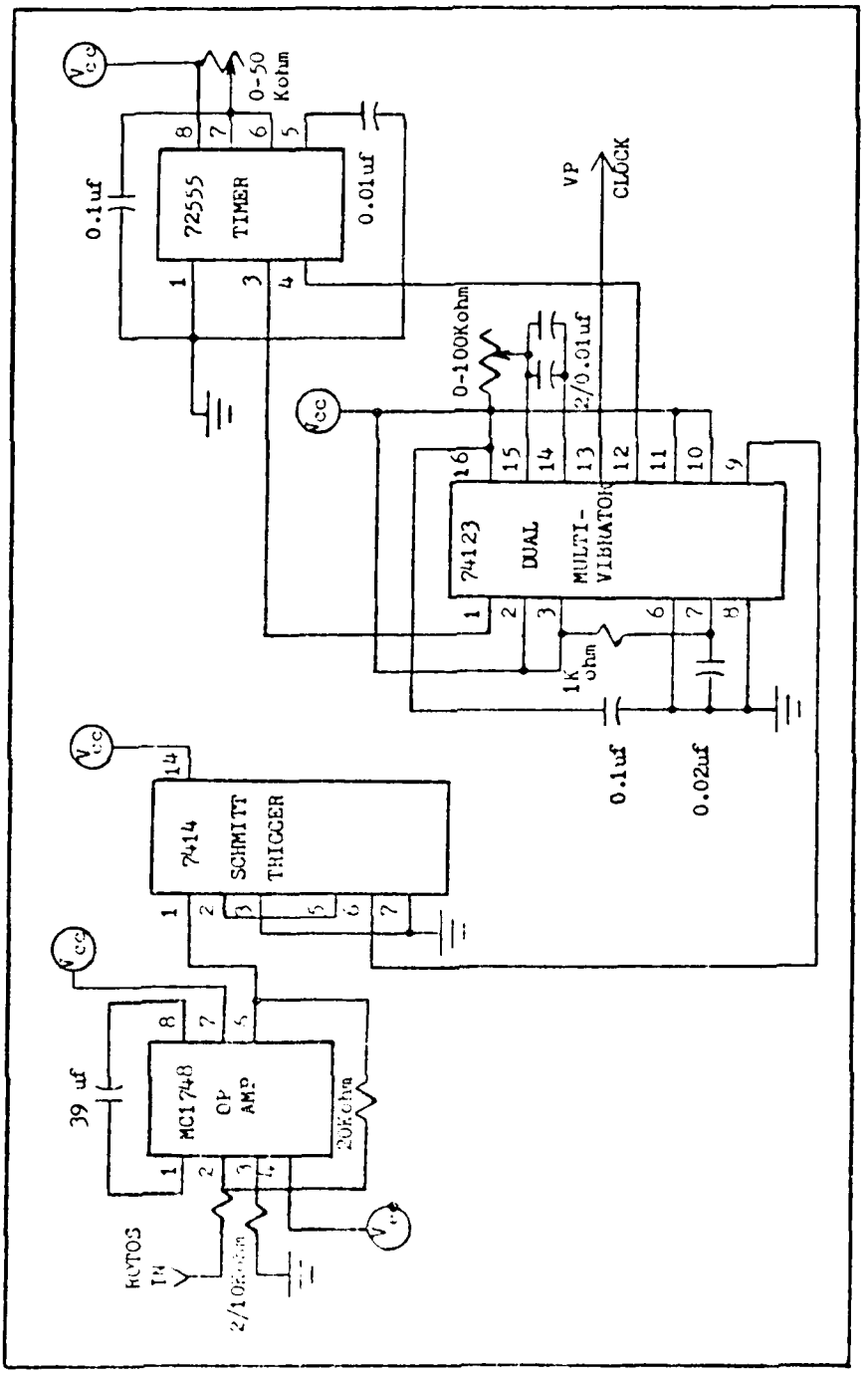


Figure 3-2

External Clock Generator, ROTOS Conditioning (18:II-6)

ing discussions this variable pulse width clock will be referred to as the VP clock.

The time delay introduced by the timer IC permits positioning of the VP clock within the sampling window of each row of electrodes on the AFIT array. The variable pulse width feature was added to the External Clock Generator circuit to make the VP clock compatible with the MASSCOMP 500 internal clocks (converter pulses). It was necessary to vary the pulse width of the VP clock to permit the MASSCOMP to generate converter pulses equal to the number of channels being sampled. This modification required that a 0 - 100 kiloohm variable resistor be connected across terminal pins Rext and Vcc on the multivibrator IC. Figure 3-3 shows how the VP clock is positioned and how the converter pulses appear relative to the VP clock and the data from each row of electrodes.

In addition to the ROTOS signal, the External Clock Generator circuitry was used to condition the stimulus control (Strobe) signal. This signal provided a control to start and stop the VP clock, giving a positive demarcation point between normal and evoked response test data. Conditioning of the strobe is similar to that used for the VP clock and proceeds as follows:

1. The strobe signal from the tape recorder was amplified using a Motorola 1743 Op-Amp.
2. This signal was then fed to a 7414 Schmitt Trigger which provided pulse shaping, sharpening of rise and fall times, and level conversion to

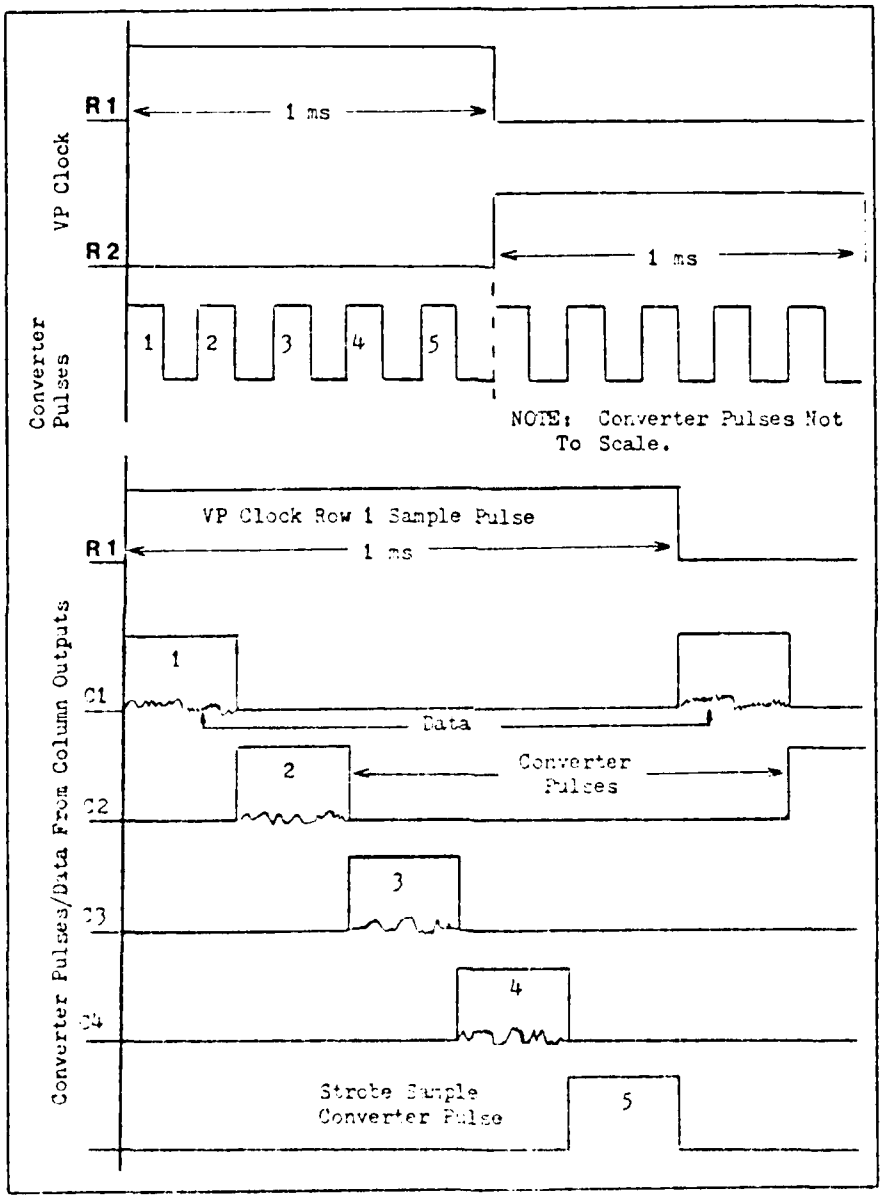


Figure 3-3

Timing Diagram Showing Row 1-4 Sampling Pulses with Reference to the Data

between 0 and +5 volts.

As a result, the TTL square wave strobe pulse which was formed could be used to control data collection. This portion of the circuit is illustrated in Figure 3-4.

Separation of the Normal and Evoked Response Cortical Data

Separation of the normal and evoked response cortical data into separate records was required to accomplish one of the primary objectives of this thesis. That objective was

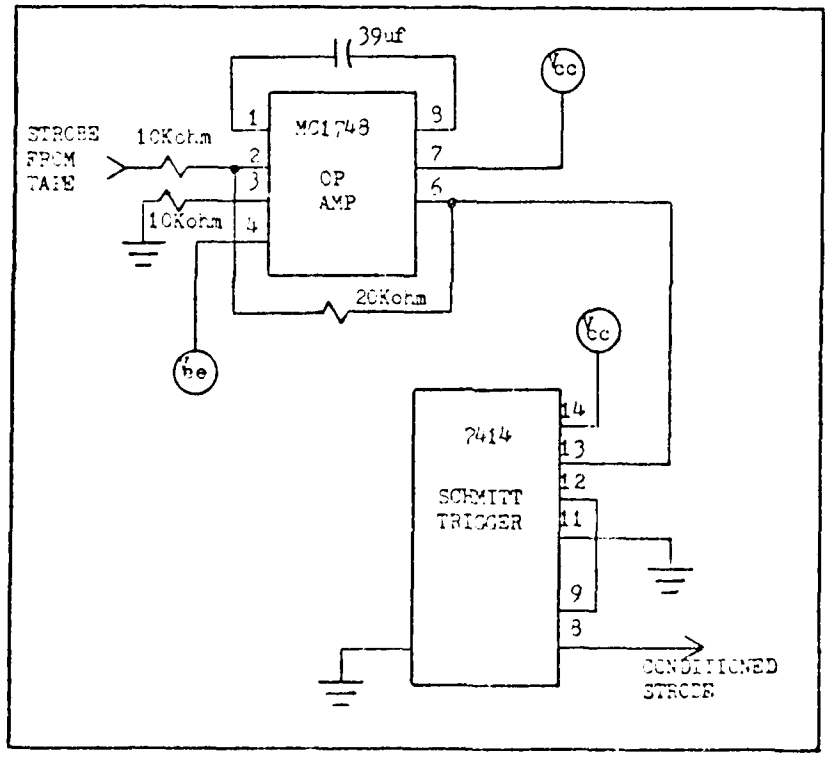


Figure 3-4

External Clock Generator, Strobe Conditioning (18:II-6)

to compute the visual evoked response from the data collected on the surface of the cortex, in order that conclusions may be drawn as to the presence and operation of the basic computa-

tional elements (BCE's). A circuit was designed and built which used the leading edge of the strobe signal to start and stop the VP clock (see Figure 3-5). Basically, during the sampling of cortical data the appearance of the first strobe pulse indicates the end of the normal brain activity. The strobe is then used as a control to turn off the VP clock signal completing the collection of the normal brain data for one test. Sampling of the normal EEG data occurs as follows:

1. When the strobe, which is connected to the clock input of a 7474 Dual "D" Flip Flop, is off, a high (Logic "1") appears on the "not Q" output of the flip flop.
2. The high signal is sent to the control input of the tri-state gate causing the VP clock to pass to the MASSCOMP Processor.
3. Sampling is completed when the strobe starts forcing the "not Q" output low (logic "0"), causing the tri-state gate to cutoff, blocking the VP clock.

Then to sample visual evoked response data the strobe start turns the sampling clock on. VER sampling occurs as follows:

1. Prior to strobe start the "D" flip flop "Q" output is low and the tri-state gate is cutoff.
2. When the strobe begins the "Q" output is forced high.
3. This high is received by the tri-state gate

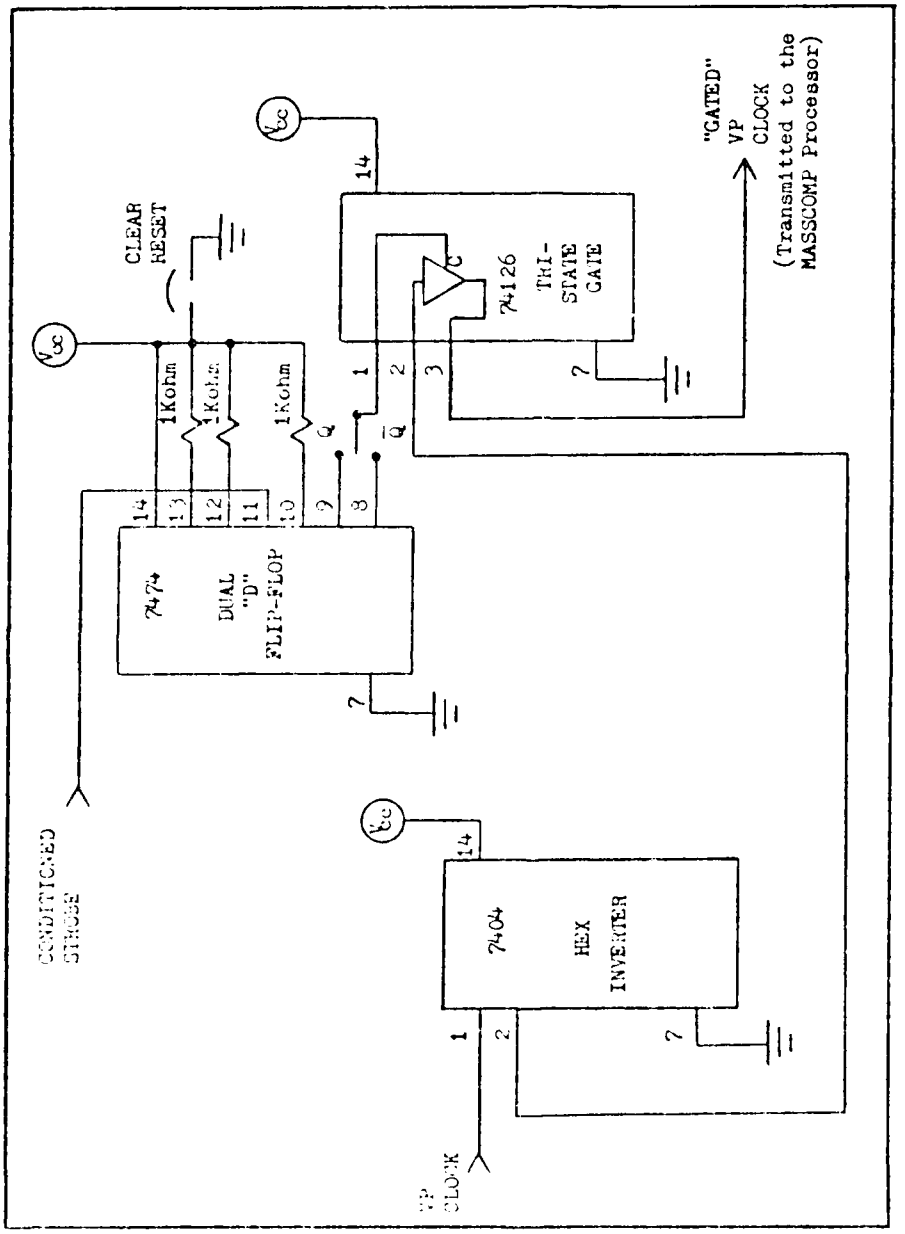


Figure 3-5. VP Clock Control Circuit

being tested for each data collection.

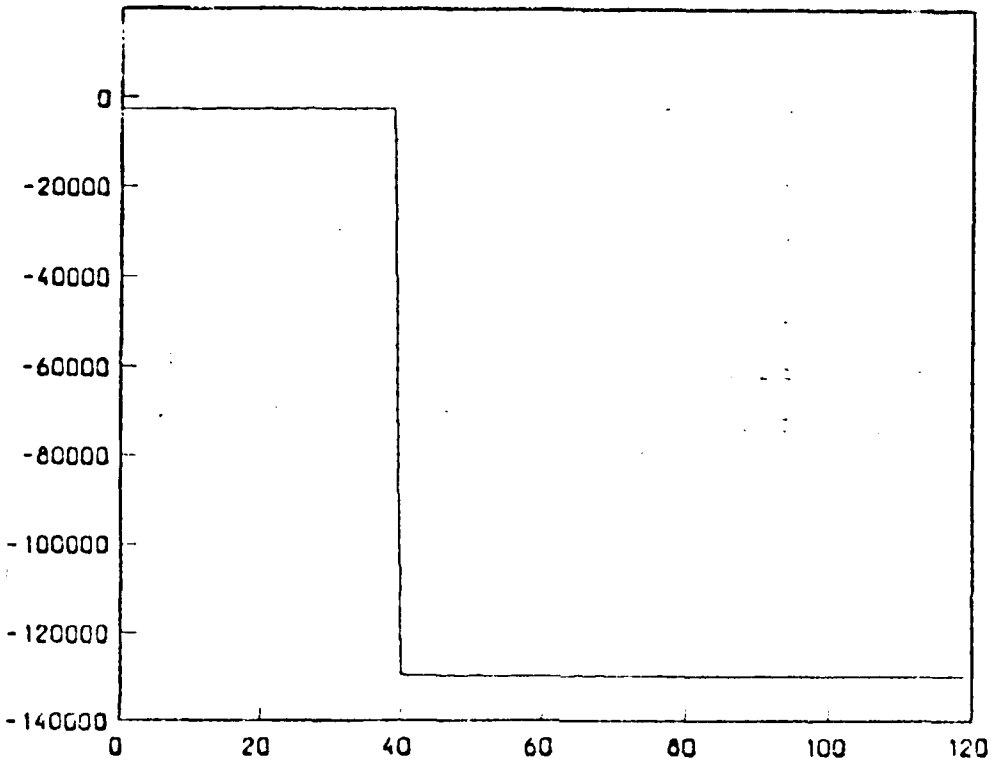


Figure 5-2. Day 3 Strobe Summed

The Third Data Collection

The third data collection involved collecting each row twice, separating out one of the data channels and the strobe channel from each collection, and running the summation program on each to compare against one another. If the output waveforms did not look similar, another data collection had to be attempted after repositioning the VP pulse in an attempt to produce better repeatability. A good position meant an area that remained fairly flat and stable throughout the test. The repeatability problem was thought to be caused by a slope

the clock width had to be adjusted to show five triggers (conversion pulses) on the second oscilloscope.

Once the data were collected, the array.out program was again used to separate the column data from the row data files and the strobe data. While the strobe data were being separated, the array.out program was made to print sample values onto the screen in order to tell what values were associated with an `on' and an `off' strobe. Once these values were determined, then the strobesum.c program was modified to recognize a strobe pulse.

Care had to be taken to ensure that the summing program summed the files correctly. To check this, the strobe file was used not only as the trigger file but also as the data file. Figure 5-2 shows the output waveform from this procdure. The waveform is a square pulse, just as expected.

Summation started on the separated files. Strictly through chance it was discovered that data collected at two different times did not produce the same summation waveform. This led to the making of more strip chart recording of sampled data. The strip chart recording proved that the repeated samplings were not digitizing the data in the same manner, when in theory they should have been. This lack of repeatability was at first thought to be caused by the tape recorder. But, through a test in which the time delay pulse was repositioned on the data, a pretty close duplication of output was obtained. This breakthrough meant that all the data had to be recollected again with the duplication criteria

Table 2

Showing the Collections and Files Made for Day 3 and Day 7

DAY 3 (differential testing only)

NORMAL TEST		VER TEST	
collection	files made	collection	files made
day3dfnmir1	day3dfnmir1c1-c4	day3dfevrr1	day3dfevrr1c1-c4
day3dfnmir2	day3dfnmir2c1-c4	day3dfevrr2	day3dfevrr2c1-c4
day3dfnmir3	day3dfnmir3c1-c4	day3dfevrr3	day3dfevrr3c1-c4
day3dfnmir4	day3dfnmir4c1-c4	day3drevrr4	day3dfevrr4c1-c4

DAY 7

(differential and absolute testing)

DIFFERENTIAL TESTING

NORMAL TEST		VER TEST	
collection	files made	collection	files made
day7dfnmir1	day7dfnmir1c1-c4	day7dfevrr1	day7dfevrr1c1-c4
day7dfnmir2	day7dfnmir2c1-c4	day7dfevrr2	day7dfevrr2c1-c4
day7dfnmir3	day7dfnmir3c1-c4	day7dfevrr3	day7dfevrr3c1-c4
day7dfnmir4	day7dfnmir4c1-c4	day7dfevrr4	day7dfevrr4c1-c4

ABSOLUTE TESTING

NORMAL TEST		VER TEST	
collection	files made	collection	files made
day7abnmir1	day7abnmir1c1-c4	day7abevrr1	day7abevrr1c1-c4
day7abnmir2	day7abnmir2c1-c4	day7abevrr2	day7abevrr2c1-c4
day7abnmir3	day7abnmir3c1-c4	day7abevrr3	day7abevrr3c1-c4
day7abnmir4	day7abnmir4c1-c4	day7abevrr4	day7abevrr4c1-c4

df - differential testing
ab - absolute testing
nml - normal testing (no strobe present)
evr - evoked response testing (strobe present)
r# - row collected
c1-c4 - columns one through four separated out

in Chapter IV and Appendix D. The array.c program was then used to demultiplex all the row data files. Table 2 shows that it is possible to collect 64 files for just day 7 alone, thus illustrating the magnitude of the data reduction effort.

The next step was to run the demultiplexed files out through a D/A board to a strip chart recorder to show the analog representation of the EEG signals at each electrode. Da.command (given in Appendix D) accomplished this task. Some of the results are given in Appendix B.

Work then commenced on writing a summing program which would atttempt to bring out any evoked response. A summing program was written which summed a specified length of data (i.e. summed each 250 length block with one another). This assumed the strobe flashed at exactly 1 Hz, and that the sampling was done at exactly 250 Hz. Upon rechecking the strobes for different days, it became evident that the assumed strobe frequency was not accurate. Thus, it became obvious that the strobe signal would have to be collected along with the four data channels, and that the summation program would have to be designed to recognize the beginning of a strobe pulse in order to sum the proper record length.

The Second Data Collection

The four data channels had to be recollected simultaneously with the strobe. This involved building the interface box (see Chapter III) to transmit the data to the A/D board. Test.multicommand had to be altered to indicate 5 channels and

by the A/D or D/A converter, denoted by the letter "C".

3. Source input, denoted by the letter "s", provides the capability to count an external clock signal.

4. A clock ground, denoted by the letter "G".

The VP clock is connected to "c" and ground on the second clock. The output is taken from terminal "1" and inserted into the clock input on the AD12F Module (A/D Converter).

Verification that the number of converter pulses is equal to the number of channels being sampled is performed with a dual channel oscilloscope. One channel is connected to terminal "c" on clock two, while the other channel is used to monitor the number of converter pulses present within the VP clock pulse width of the VP clock.

First Data Collection

The data were collected using the MASSCOMP Quick Choice option to digitize the desired row signal from all four data channels and to store it onto another file. For each day of testing, there was a normal and a VER test which had to be collected. Also, during the latter days an absolute test was also conducted. This made a total of four tests per day, or sixteen data collections per day (see table 2). While data collection took place, efforts were made to develop a program which would separate out the four column signals from each row of data. This became the array.c program which is described

accessed on the front panel, while two software controlled clocks are internal. In order to sample "n" channels the VP clock must be set to allow "n" converter pulses to appear within its pulse width, as shown in Figure 3-3.

The cable connection configuration of the VP clock and the MASSCOMP converter pulse clock are shown in Figure 5-1.

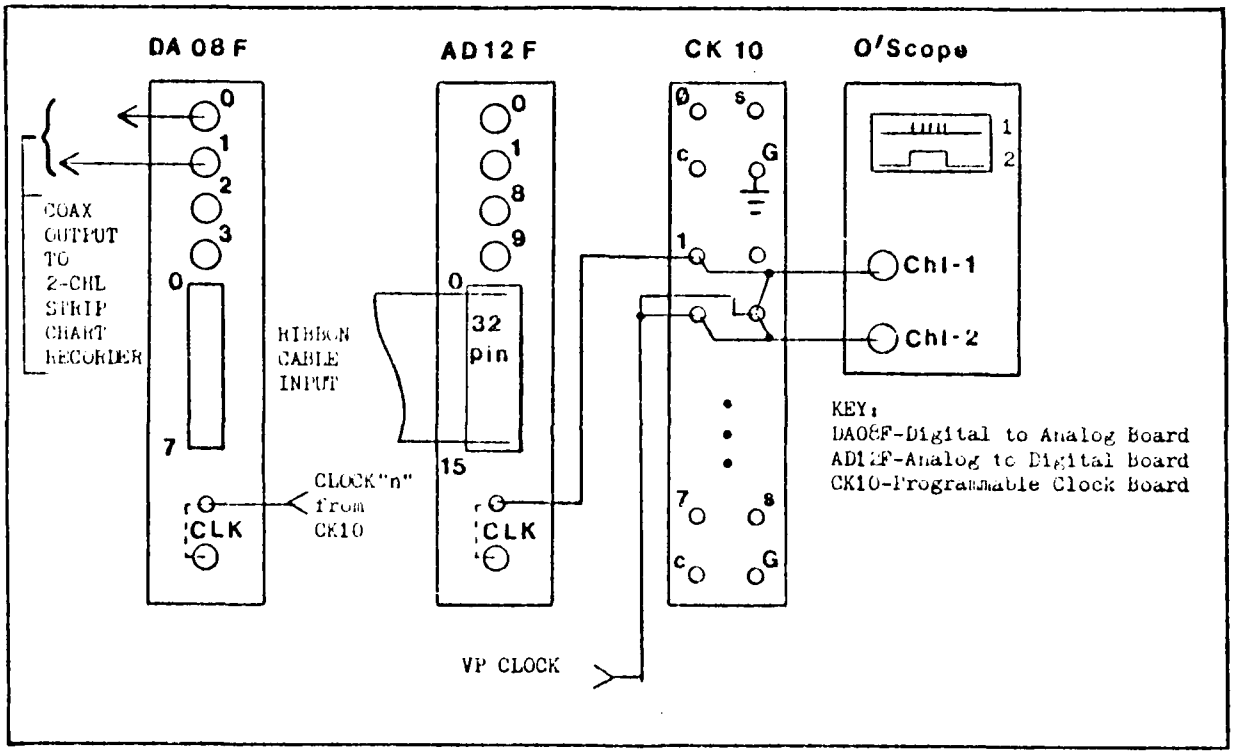


Figure 5-1. Cable Connection Configuration of the VP Clock and the MASSCOMP Converter Pulse Clock

Each clock on the MASSCOMP clock board consists of four terminals:

1. A normal output, denoted by a number from 0 to 7.
2. Control input accepts an external clock which determines the number of converter pulses required

V. Data Reduction

Initial Steps

The first steps in analysis of the data focused on four main areas:

1. Reviewing the tapes and making a log of what was done based on the voice track (see Appendix A).
2. Studying Hensley and Denton's thesis in order to understand how the data were collected.
3. Studying Hayes' thesis and his circuit to understand his analysis effort.
4. Learning about the MASSCOMP MC-500 computer.

The MC-500 computer is more flexible when it comes to data acquisition than the Eclipse. Because of this, it was decided that very little of Hayes' circuitry would be needed. The circuit that is described in Chapter III was designed for the sampling/digitization phase, and it allowed for distinguishing between normal data and strobed (VER) data.

MASSCOMP 500 Multichannel Sampling

Before the A/D conversion of the data could begin it was necessary to set up the processor in the multichannel sampling configuration. This configuration required two clocks. The first was the VP clock generated from the ROTOS as discussed in Chapter III. Clock two came from the MASSCOMP CK10 module and provided the converter pulses to sample the data channels. The CK10 module provides 10 programmable clocks, eight are

varisum.c was generated (See Appendix D). The results from this test are given in Chapter VI.

Digital-to-Analog Conversion

To show that EEG signals are collected by each electrode, the demultiplexed digitized signals have to be sent through the Digital-to-Analog (D/A) converter to a strip chart recorder. The command that does the conversion, da.command, is given in Appendix D, and samples of the strip chart recordings are shown in Appendix B.

channels the file contains. In this case, five channels were usually sampled (four data channels and one strobe channel). The demultiplexed channels can be plotted on the graphics terminal screen and the data stored in the file specified.

VER Extraction

Once a data and a strobe file are collected the appropriate summing program can be used. If an unfamiliar strobe is being used, the array.c program should be modified to show the strobe values. This can easily be done by taking off the comment marks (/**/) on line 140 and recompiling the program.

(The line will read:

```
printf("gpbuff[ %d ] =  %d  \n",count,gpbuff[count]);
```

Once the character of the strobe is determined, the strobesum.c program can be modified so it will recognize an "on" strobe pulse. This is very important so that the data start summing at the same position each time with respect to the strobe pulse.

The next step in the thesis effort was to analyze how varying the sample size effects the VER waveform. According to Hayes, "The form of the response is dependent on the number of epochs averaged, where one epoch is the data recorded between stimuli" (18:V-2). In order to prove or disprove this statement, it was decided to use the data already collected and to only vary the number of summations actually performed by the summing program. For this modification, the program

simply the speed at which the A/D converter is triggered. For multichannel transfer, two clocks must be used. One clock generates the sampling rate (the interval between frames) and the other determines the actual converter speed.

When allocating file space on the disk for each transfer, the following formula should be used:

Disk Space =

Samples in a frame x Sampling rate x time x 2 bytes

Analog-to-Digital Conversion

The Quick Choice method was utilized to perform the A/D conversion or digitization. The data were collected by rows using the multichannel sampling option. Referring back to Figure 2-3, the data had four rows multiplexed within each output channel (or column). By moving the time delay detector via a potentiometer, the four rows can be singled out for sampling by the test.multicomm command. This command is explained in Appendix D. Care needs to be exercised to ensure that the time delay trigger is positioned near the middle of each row section of data and that the signal is as horizontal as possible. If this is not done, faithful reproduction of the data waveform is not possible. Once a row is sampled and stored, it can then be demultiplexed into the individual column electrode signals using the compiled array.out algorithm. The C-program,array.c, is given in Appendix D. This program takes the multiplexed file obtained using the multichannel sampling command, test.multicomm, and separates it into however many

Choice allows the operator to interactively specify the commands that would otherwise be typed directly into the UNIX command interpreter or "shell". A good explanation on how to use Quick Choice is in the MC-500 Quick Choice Users Manual (43).

Data Transfer

Data may be transferred between the input channels and the system disk by way of memory buffers. In a fully buffered transfer, data are not stored to disk until sampling is complete. This method is limited by the amount of available memory and contiguous file space. In a partially buffered disk transfer, data are also placed in memory buffers, but the buffers are written to disk while the transfer is still in progress. This method avoids memory limitations, but the transfer rate is still limited by the speed of the disk. Partially buffered transfers are preferable if one is transfering large amounts of data (i.e. > 200,000 samples); fully buffered transfers are preferable for higher speeds.

The Clocks

The CK10 module contains ten independent clocks. Eight of the clocks have connectors on the front of the card. A single clock can be used to trigger sampling. Multiple clocks can be gated together to create complex waveforms.

Transfer Rates and Sampling

For single-channel sampling, the data transfer rate is

IV. The MC-500 and Software

The MASSCOMP MC-500 computer was the only computer used in this thesis. The MC-500 is a 32-bit unix-based mini computer system which allows three dedicated subsystems to perform high speed computation, graphics display, and data acquisition simultaneously. The computer will accept Fortran, C, and Pascal programs, although C is the preferred language. It has a 1-MHz 68010 CPU which works with a 10-MHz 68000. Its 8-slot back plane Multibus handles 6 Mbytes/s. It also has 16 Mbytes of virtual memory (43:112-113). For this thesis, the system was configured with a 27-Mbyte Winchester drive and a 700-KByte, 5 1/4" floppy disk drive. An upgrade is planned which will increase the hard disk storage to 166-Mbytes and add one magnetic tape drive. Appendix F contains more details on the MASSCOMP MC-500's specifications.

Quick Choice

Software functions are controlled through a program called Quick Choice (the MC-500 menu system), by direct access using the UNIX command interpreters (the C shell), or by user written programs. This thesis used the Quick Choice option to digitize the data. C programs were also written to demultiplex the data and compute the evoked response. Quick Choice allows the operator to specify actions needed through a series of choices given in what it calls a "menu". Essentially Quick

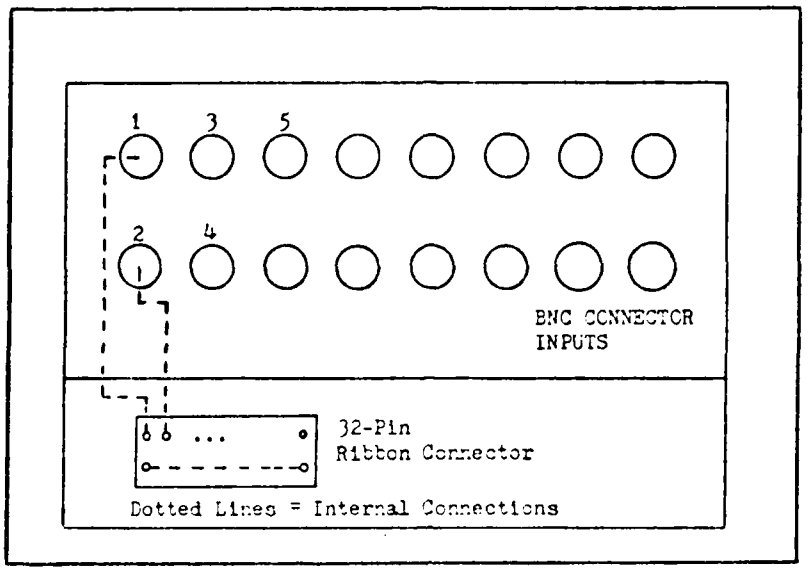


Figure 3-6

Block Diagram of Incremental Interface

MASSCOMP A/D board. Collection of the data from the tape recorder for this thesis required only five inputs, the four column outputs from the array and the strobe signal. However, the capability exists to install 16 BNC connectors should future theses efforts require it.

control input, allowing the VP clock to pass to the MASSCOMP Processor.

In addition to the circuitry discussed, a 7404 Hex Inverter is required to change the VP clock to a negative going pulse, so it is compatible with the MASSCOMP internal clocks. Finally, the "D" flip flop must be cleared prior to the start of each test. In the breadboarded configuration this is accomplished by shorting the "clear" pin to ground through a "normally open" push button. The specifications for the integrated circuits used in all the previously discussed circuits are contained in Appendix C.

Incremental Sampling Interface

Actual transfer of data from the multichannel tape recorder to the MASSCOMP Real-Time Processor required the construction of an interface. This interface will perform the physical conversion of the transmission medium from coaxial cable from the tape recorder outputs to the ribbon cable input on the A/D converter board (See Figure 3-6).

The interface was constructed to accommodate the connection of 16 coaxial cables with BNC connectors to receive the output from the tape recorder. An internal connection was then made from the BNC connectors to a 32 pin ribbon connector. One side of the ribbon connector receives the 16 signals from the BNC connectors, while the 16 pins on the other side are a common ground terminal. A ribbon cable transmits the data to the incremental or consecutive sampling input on the

in the data resulting from changing capacitance in the array JFETs.

Upon collecting day 7 data, it became apparent that Hensley and Denton had changed their strobe to a different type. The strobe turned out to be a quick spike instead of a square pulse. This meant changing the strobesum.c to recognize the values associated with this new strobe. This change also had to be validated using the same procedure as before. Figure 5-3 shows the new strobe being summed while using the new

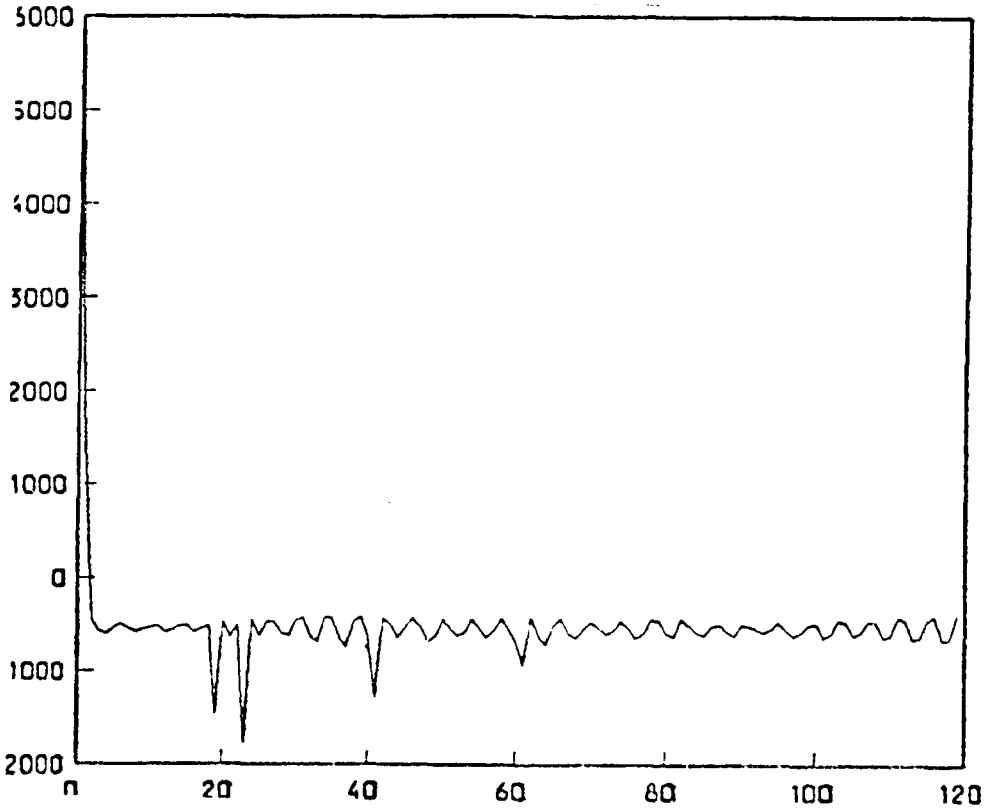


Figure 5-3. Day 7 Strobe Summed

strobe as the trigger. The quick spike is very evident. After this, each days strobe was checked to make sure that the

values in strobesum.c were still good. The plots from the summation program are given in Appendix C. All the data collected and demultiplexed during this last phase of data collection are stored on 5 1/4inch floppy disks. Appendix E contains a table listing the floppy disks and what is recorded upon them.

The last testing which needed to be performed was using the varisum.c routine to test the effect which the number of summations has upon the output waveform. The data which had been digitized and summed previously were used in this program with only the number of summations changed. The results of this test are discussed in Chapter VI.

VI. Results and Analysis

Summary

Investigations were carried out in three areas on the data collected from the multielectrode array implant experiment, performed by Hensley and Denton (19) in October and November 1982. Initially, hardware was designed to condition and control the timing and sync signals. Next, software was developed to digitize and demultiplex the 16 signals from the AFIT array. Thus, each electrode signal could be converted to an analog file for examination of EEG characteristics. Finally, a C routine was written to calculate the visual evoked response of the signals collected off the cortex surface. The results of those investigations will now be discussed.

Data Conditioning and Controlling Hardware

The initial area to be addressed, as outlined in Chapter I, was to prepare the data for digitization. This was achieved through the construction of hardware which generated synchronization and timing pulses (See Chapter III). Sync signals were developed which set up a demarcation between the normal EEG data and the evoked response data which allowed the two types of data to be stored in separate files. A sampling pulse (VP clock) was constructed from the row one control signal (row one turn-on signal, ROTOS) which sampled five channels, permitting simultaneous A/D conversion. Finally, an

interface was fabricated to convert from a coaxial to a ribbon cable transmission medium. This facilitated the simultaneous digitizing of five data channels. Thus, the hardware constructed supplied the necessary sampling signals for the correct operation of the software.

Analog EEG Recordings

As a result of the software developed, reduced data were recorded in two forms. The first set to be discussed is the demultiplexed data from each array electrode which appears in analog EEG format. Numerous strip chart recordings were made by performing a digital-to-analog conversion on the files from each electrode. A typical analog recording is illustrated in Figure 6-1.

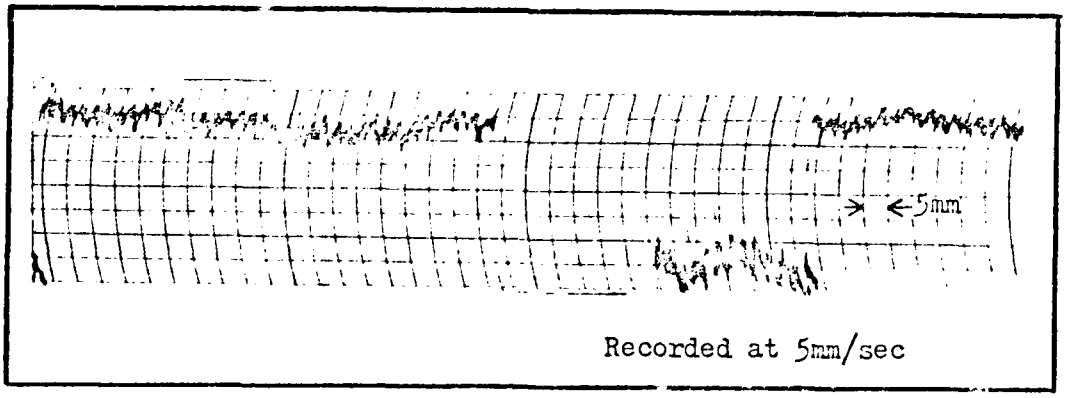


Figure 6-1. Day 3, Row 3, Column 4 Electrode Recording

The data illustrated in the above figure appears to contain bursts of seven Hz alpha waves. The appearance of the alpha waves seems to indicate that the array is a suitable method of collecting information from the cortex surface. Additional recordings are contained in Appendix B. Although,

alpha frequency waves were found in the early days of testing, recordings of later tests, day 6 and beyond, showed some undersireable effects.

Some of the data appeared to be contaminated by the strobe stimulus signal and 60 Hz interference. Recordings from day eight seem to show that the strobe signal was introduced within the cortical data channels. This anomally may have been caused by the close proximity of the strobe circuitry to the array output circuitry. Additionally, the large magnitude of the strobe (0 to -6 volts) was another factor which may have caused its appearance in the data. This is especially prevalent in data collected after day 7 when the strobe was changed to a short duration high magnitude spike. The effect of 60 Hz interference can be observed in data collected from day 6 on. However, this interference appears to be uncorrelated and has little effect on the evoked response calculations.

It is also interesting to note that the presence of 60 Hz interference was most prevalent in the data collected in the differential mode (See Appendix B-6). Data collected in the absolute mode was measured from a row output amplifier to a reference electrode. Differential data was measured between two row output amplifiers. The data collected in the absolute mode for day 9, row 1 column 1 shows no discernable presence of thestrobe. Therefore, perhaps the lack of a positive ground in the differential mode allowed the strobe to leak into the cortical data.

Another anomaly observed in the analog data caused noticeable variations in the evoked response calculations. This was the lack of repeatability in the analog data. In other words, recordings made of data taken from the same data sampled twice showed marked differences (See Appendix B-8). Although these differences appeared as minor perturbations on the stripchart recordings, the VER plots showed marked differences. As mentioned previously, by shifting the VP pulse to a more stable (flat) area on the data signal the effects of these pertubations were minimized.

One final observation regarding the analog EEG data would seem to provide information as to the state of deterioration of the array. Data recorded from the row 3, column 1 and 4 electrodes collected on day 11 seems to show only strobe and 60 Hz interference (See Appendix B-7). Examination of a photo taken thru an electron microscope of the multielectrode array (19:83) after it was removed showed significant degradation of the array electrodes. In fact, the electrode at position row 3, column 4 is missing in the photograph. Therefore, the lack of any noticeable brain activity on the day 11 strip chart recording may indicate array degradation started sooner than was thought by Hensley and Denton. This subject is addressed again in the following section.

VER Waveforms

First, it will be noted that the methods and the routines used to extract the VER waveforms were validated bycollecting

and summing a known waveform (i.e. a sine wave). The next area of concern dealt with the number of samples which were collected in each test. Theoretically, a specific sample size could be derived using statistics. Realistically, this method should be used if each sample cost the experimenter something, such as money or time. In this case, samples cost practically nothing. If anything, the question was not "How many samples should one take?", but more, "How many samples can one take?". Because of the memory constraints addressed previously, a sample size of greater than 200,000 was not possible using fully buffered data transfer. Partially buffered transfers had been ruled out because of increased time for collection, even though it permitted larger samples sizes.

Appendix C contains the summed VER plots for most of the days. Some of the VER plots for specific electrodes are missing because of the inability to achieve a reasonable level of repeatability when data from a specific electrode were sampled and plotted twice. Days 3-6 show VERs under the first strobe used by Hensley and Denton. When the plots are physically arranged in row and column format by each day, some observations can be made.

1. All the days show a marked decrease or negative spike at the exact same time as the strobe cut-off, indicating a leakage of the strobe signal into the data channels.
2. As the days of testing progressed,the waveforms smooth out considerably, and by day 6 the VER

plots show minimal pertubations. This could indicate that the dog was becoming accustomed to the 180 millisecond square wave strobe signal.

3. Some collective column output signals appear smoother than the rest. This is especially apparent in column 4 waveforms for days 5 and 6. This could indicate a degradation of the circuitry by the CSF in the dog's brain.

It is known that when the AFIT array was removed from the dog's brain, some of the electrodes had deteriorated. Hensley and Denton believed that the damage was a result of the extraction procedure (19:82) since the cortex area under the implant appeared normal. However, looking at how the VER waveforms changed as the days progressed indicates that the implant was probably affected by the CSF prior to removal.

From pictures in Hensley and Denton's thesis it appears that; row 4 electrodes had almost completely been stripped away, column 4 output channel and rows 3 and 4 input signal channels had begun to deteriorate, and row 1's internal circuitry had also been stripped away. What is not known is how soon this deterioration began, and what effect it had on the data collected from each electrode.

To minimize the effects of the square wave strobe on the data, Hensley and Denton changed the strobe stimulus marker to a spike waveform on day 7. They apparently believed that the square wave strobe marker could be biasing the data too much.

If the data collected while the new strobe stimulus was applied is laid out in columns and rows, it is harder to tell the spike strobe cut-off point. One also sees a sudden increase in activity in the waveforms, maybe because the dog was not accustomed to this new strobe.

On day 7, a test had been done using a metal bucket and a black cloth to shield the light from the dog's eyes. This is an interesting test because it allows summations to be performed since the strobe is present, but the data should also be absent of any evoked response. Unfortunately, only one row was capable of being collected from this test because the data was unstable. If this row is used as a comparison for the evoked response data collected, one notices in day 7 that there are usually two spikes that appear at the beginning of the plots within the first 20 milliseconds. Some of the plots do not show these spikes, but usually these plots also indicate a lessened brain activity. In fact, as with the square wave strobe, the plots show a lessening in waveform pertubations from day 7 to day 8. Also evident is a columnar correlation. The column waveforms between the two days are very similiar. Since the data were digitized by rows and not by columns, this again indicates that there may have been something affecting the column output channels from the array during the actual testing with the dog.

On day 11, the strobe's magnitude was doubled. This caused a noticeable change in most of the data except column 1 data (for both absolute and differential testing). This is

also seen in day 8's data. The fact that day 11, column 1 data also exhibits the same waveform as day 8, column 1 data signifies that there could have been something wrong with column 1's output channel at that time.

The final area of analysis covered the effect that changing the number of summations, to derive the VER, had upon the shape of the waveform. As mentioned previously, Hayes believed that varying the number of summations would greatly change the waveform because of possible non-stationaries. Using the varisum.c program the plots for 40 summations and 60 summations are shown in Figure 6-2 and 6-3, respectively. There are differences, but they are minor compared to the general shape of the waveform, which is the same in both summations. These differences in the VER waveforms are probably caused by variations in the summation of uncorrelated noise. Thus, the reliability of the methods used to extract the VER seems to have been verified. Unfortunately the length of the test only allowed 60 summations.

This variable summation test was performed on the only data which were able to be collected from an ear probe referenced to the L-shaped electrode on the array (amplifier #5). It was hoped that a plot matching Kemsley and Benton's plot (Figure 6-4) could be duplicated. Unfortunately this was not done, although some similarities are noticeable.

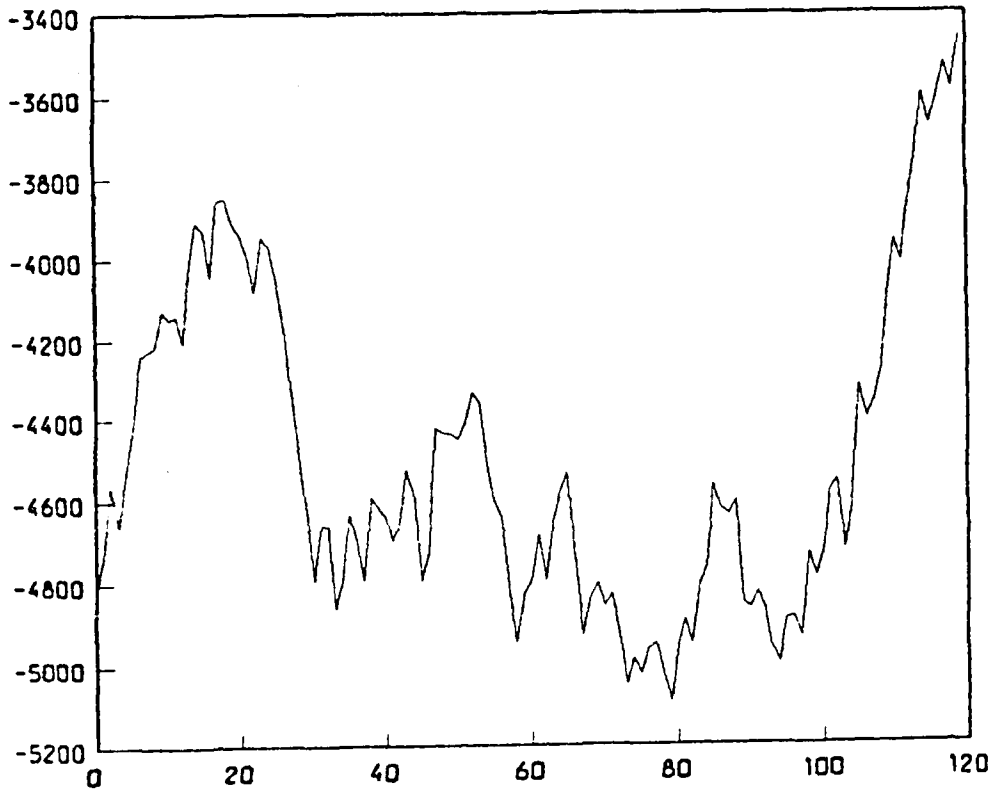


Figure 6-2. Day 11, Row 5 Using 40 Summations

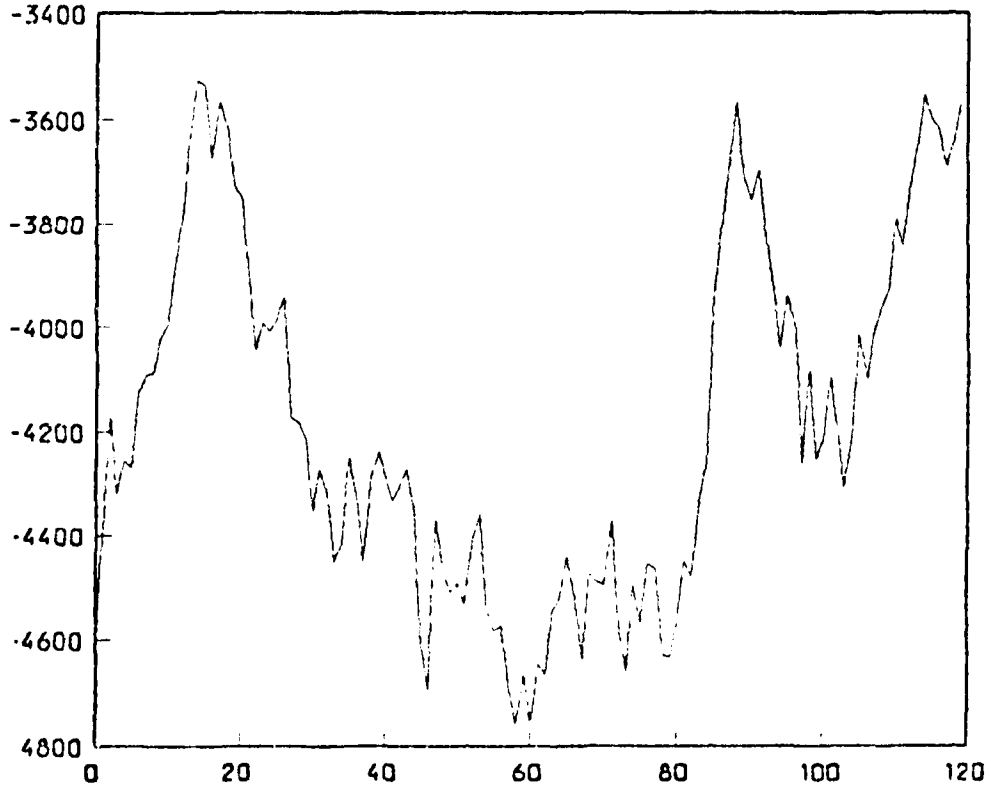


Figure 6-3. Day 11, Row 5 Using 60 Summations

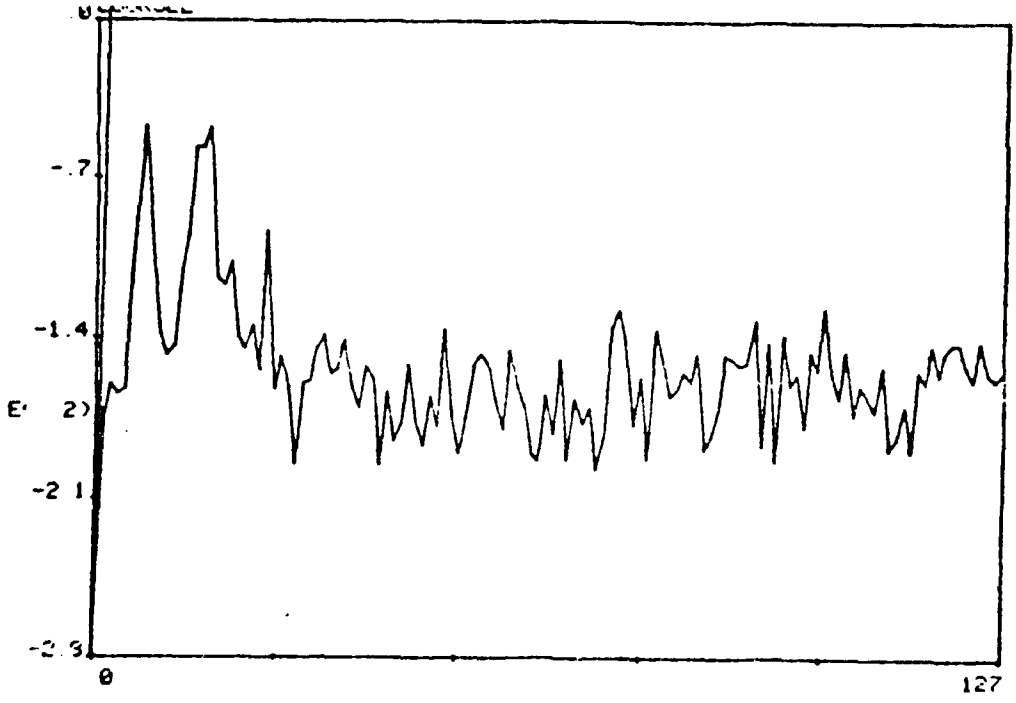


Figure 6-4. Hensley and Denton's VER Derived Between the L-Shaped Electrode and the Dog's Ear (19:A-16)

VII. Conclusions and Recommendations

Conclusions

This thesis accomplished most of what it set out to do. The EEG strip chart recordings display evidence of low frequency alpha waves, thus indicating that the semiconductor array implant is a suitable method for extracting brain data. The fact that sixty cycle hum (evident in the EEG strip chart recordings) does not appear in the VER curves confirms the theory that the summation/averaging method can effectively filter out uncorrelated noise.

Most of the data collected by Hensley and Denton was studied and comparisons made between waveforms of different days and different tests. The tests conducted using the spiked strobe of days 7 thru 11 are probably the most useful since evidence indicates that the longer strobe's signal leaked into the data collection circuitry. Unfortunately the spiked strobe was not used until well after the array was implanted. By this time the CSF in the dog's brain probably had begun to deteriorate the implant. This creates a lack of confidence to some degree in the fact that a possible evoked response was found.

This thesis has devised a method by which further experiments can directly sample the data and analyze it without the use of some recording device. By using the interface box and the digitizing and summing algorithms, near-

realtime data analysis and feedback is possible.

Final Remarks and Recommendations

None of the writing so far adequately expresses the amount of work the analysis required and the numerous frustrating problems encountered. The work and the frustration were brought on in part by the fact that this thesis followed after two other thesis efforts (Hensley and Denton's, and Hayes') which supplied the data and also the first data analysis attempt. Critical information needed by this thesis was not documented, thus much effort and time was spent in areas which should not have been required. Two major examples of the lack of documentation are:

1) No log was written down by Hensley and Denton as to what tests were done each day and no indication as to when they changed strobes and what the characteristics of each strobe were. Although there is a voice track on the tape, the information recorded there is very sketchy and one still needs to have a written log. Appendix A now contains the log for days 3 thru 11, taken from the voice track. The log does not include days 1 and 2, and 12 and 16; since data from these days could not be digitized for reasons mentioned in Chapter III, page III-1.

2) Hayes left no schematic to the circuit which he designed. Fortunately, this thesis did not need to use much of Hayes' circuitry.

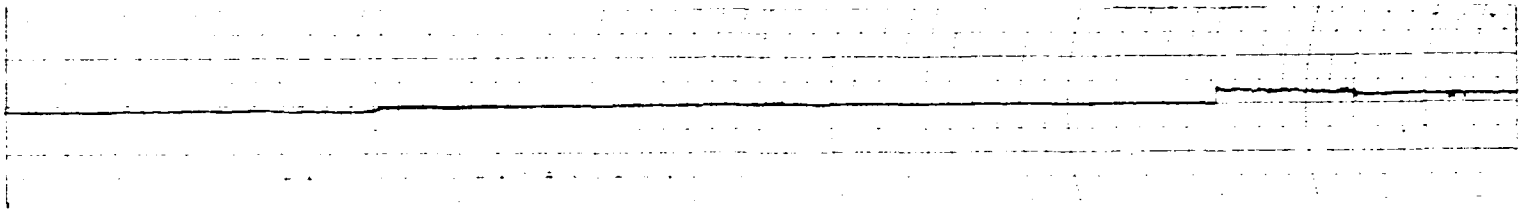
This does not necessarily mean that the two prior thesis

APPENDIX C

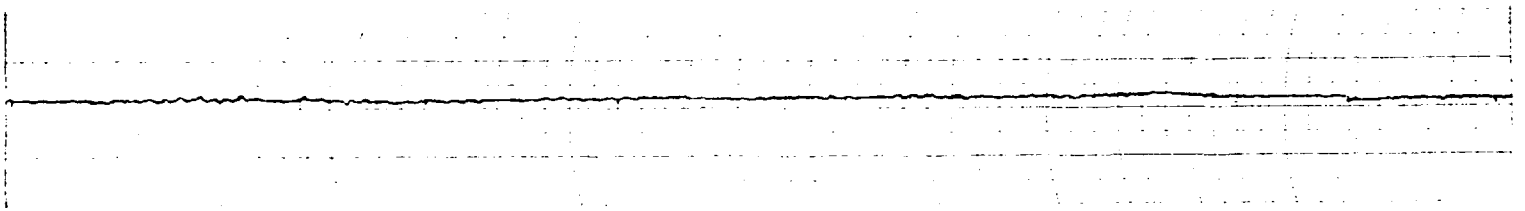
VER Summation Plots

This appendix contains the VER summation plots. The time scale along the x-axis represents 540 milliseconds. This length of time was chosen to show as much detail as possible in the waveforms. In some instances an even shorter time length could have brought out more detail, but to keep the time scale consistent, this was decided against.

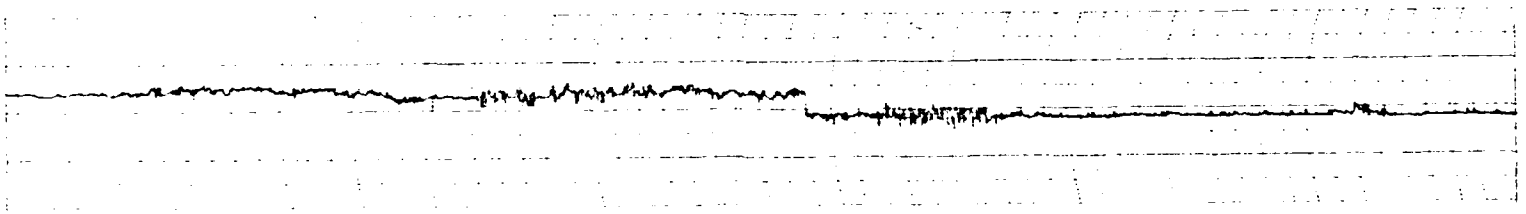
Confidence Test: EEG Data, Day 3, Row 2



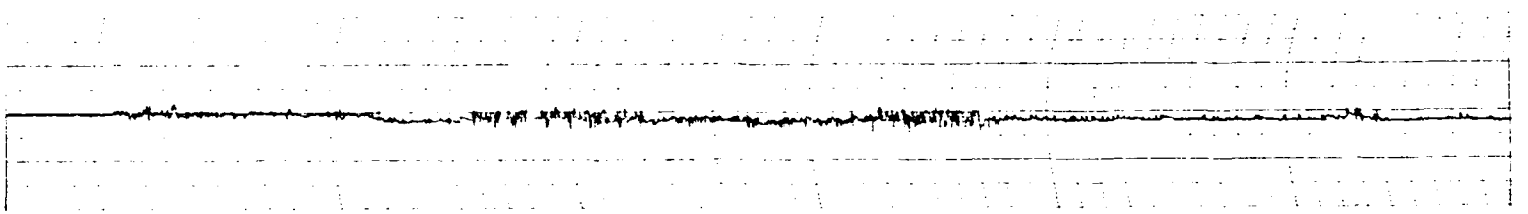
Test A, Column 1



Test B, Column 1

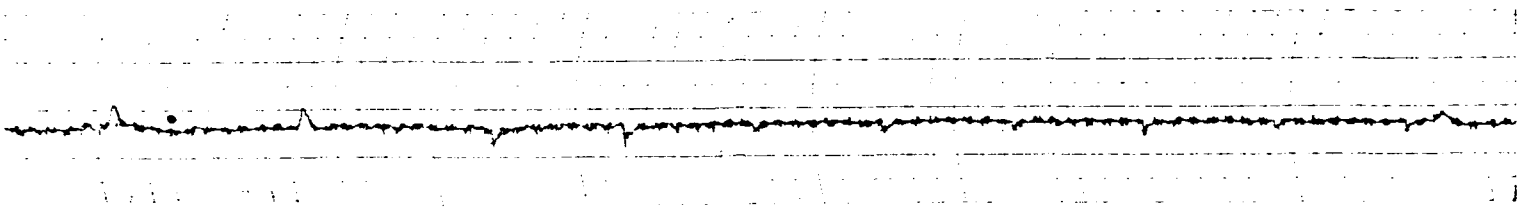


Test A, Column 2

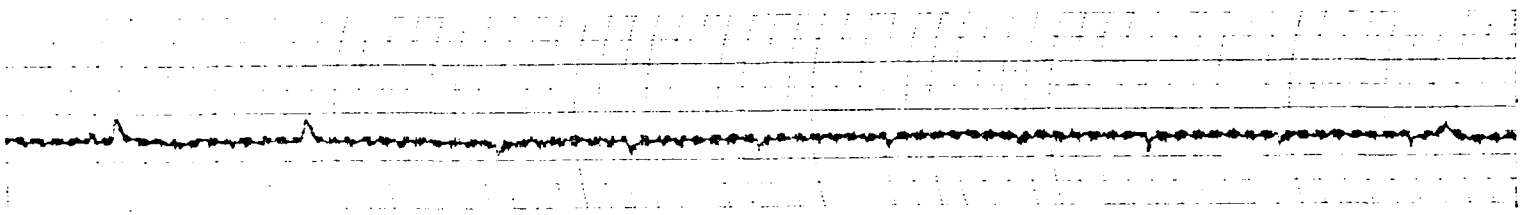


Test B, Column 2

EEG Data, Day 11, Row 3, Differential Mode

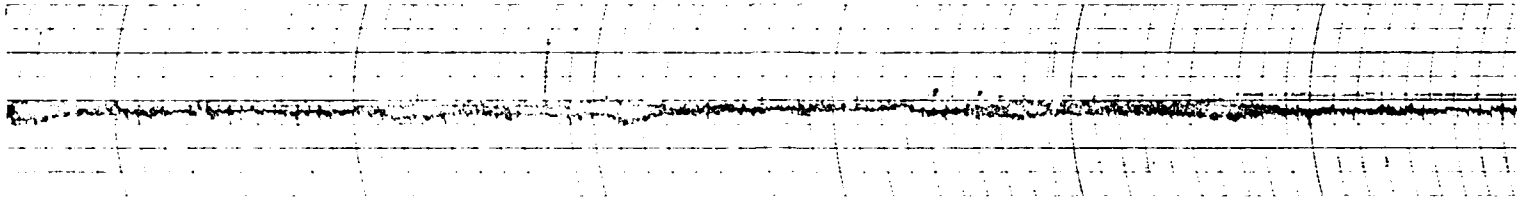


Column 1

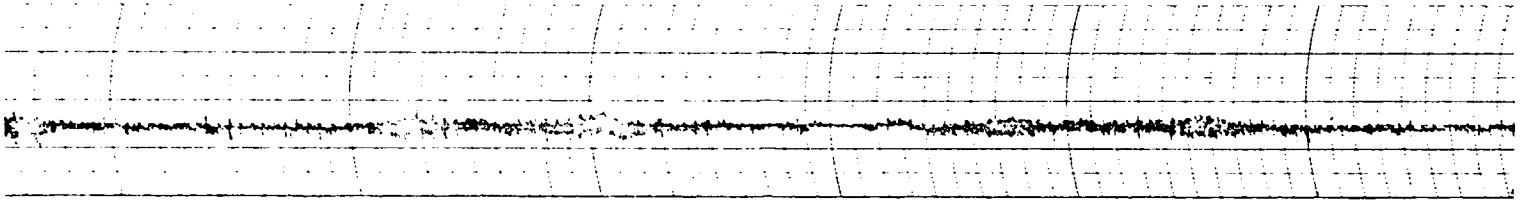


Column 4

EEG Data, Day 9, Row 3, Differential Mode

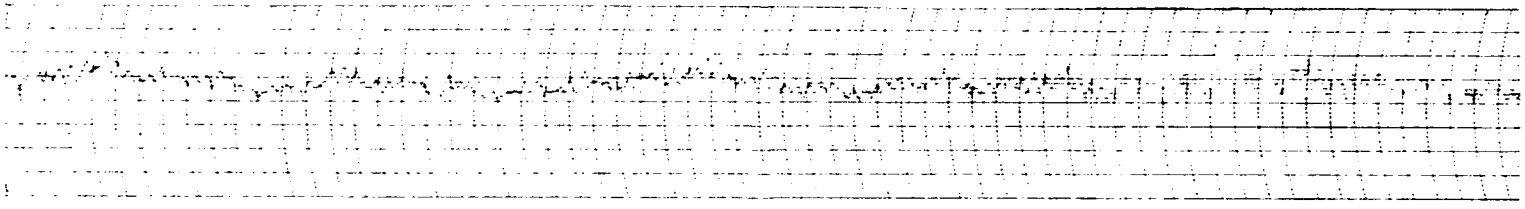


Column 2

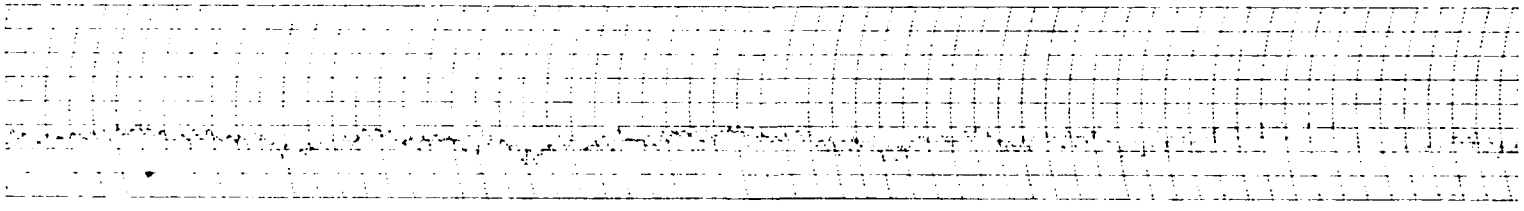


Column 3

EEG Data, Day 9, Row 1, Absolute Mode

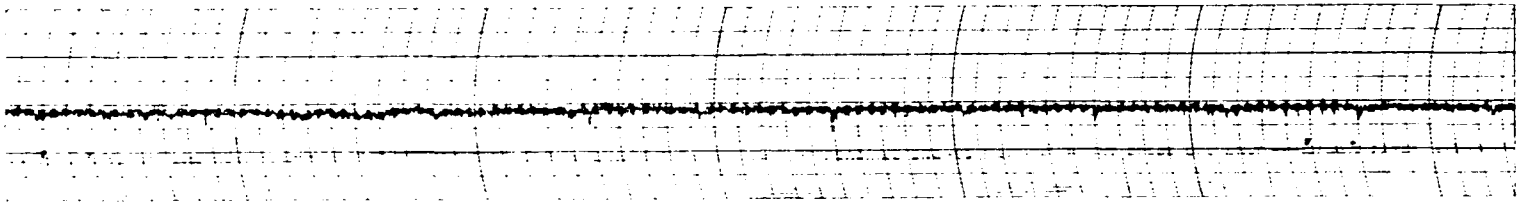


Column 3

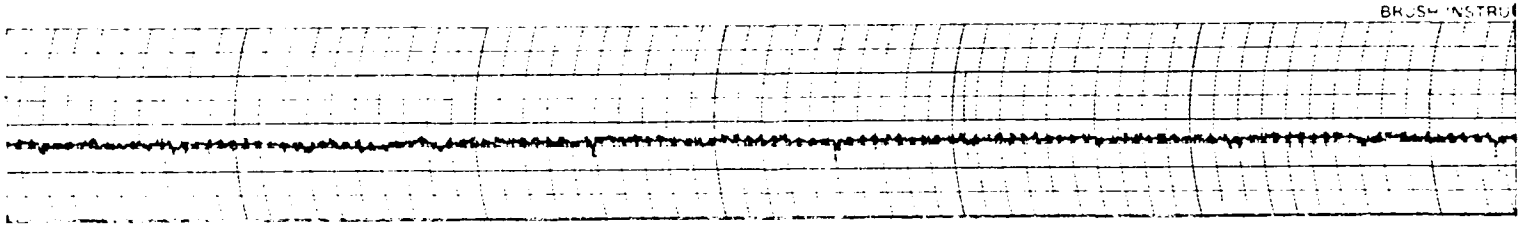


Column 4

EEG Data, Day 8, Row 3, Differential Mode

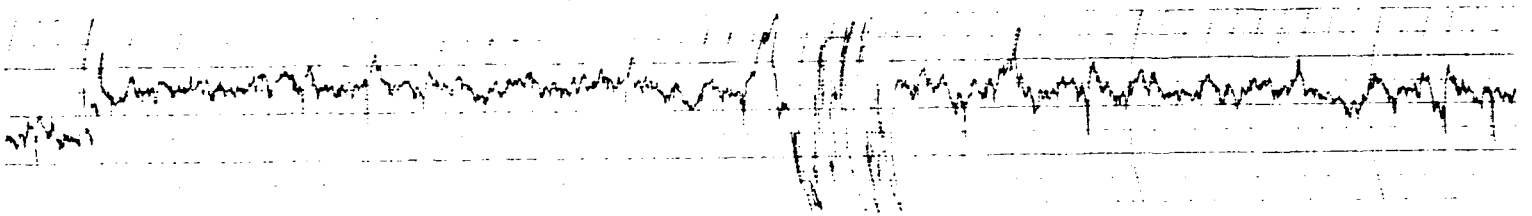


Column 2

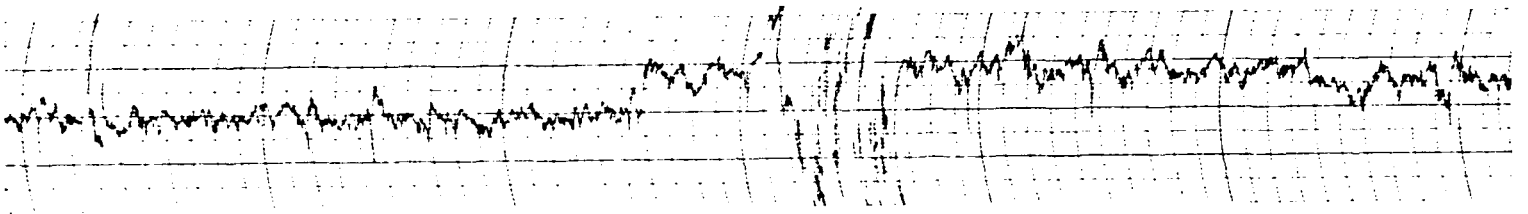


Column 3

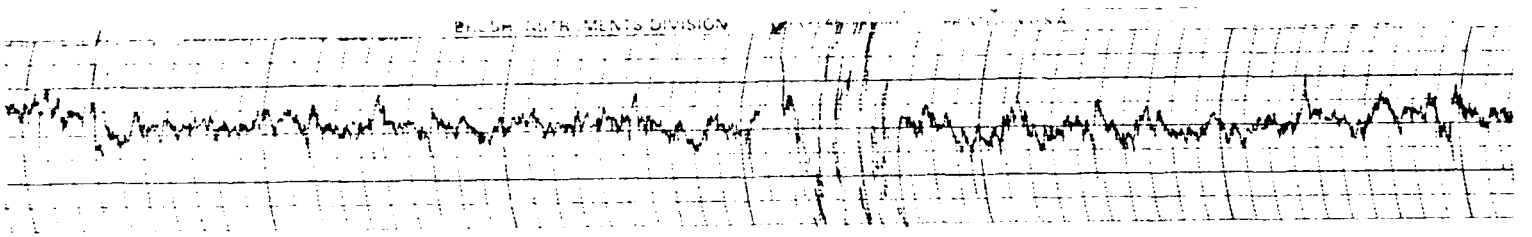
EEG Data, Day 7, Row 3, Differential Mode



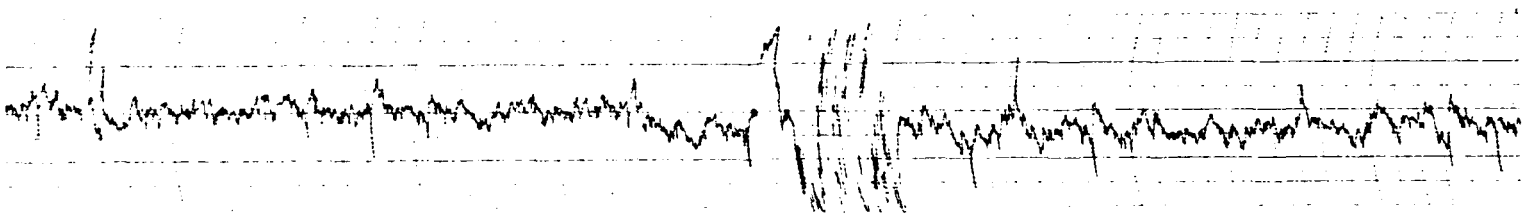
Column 1



Column 2

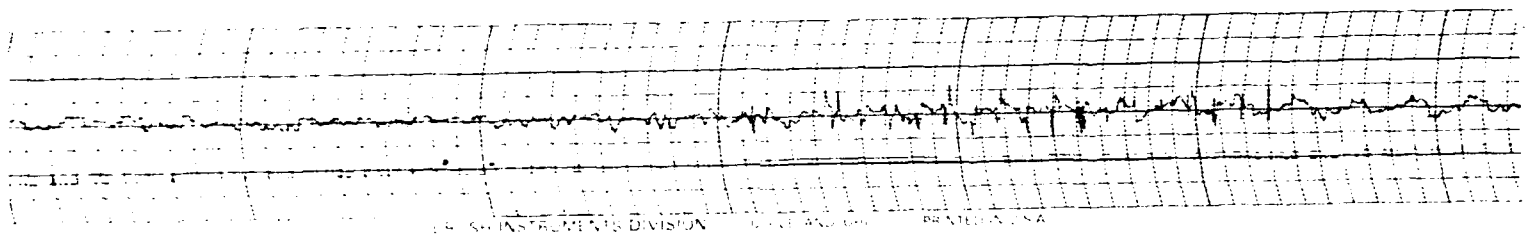


Column 3

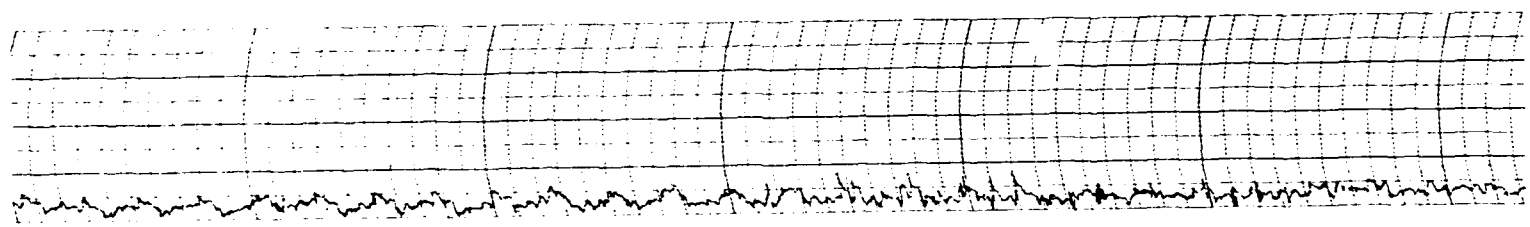


Column 4

EEG Data, Day 6, Row 3, Differential Mode

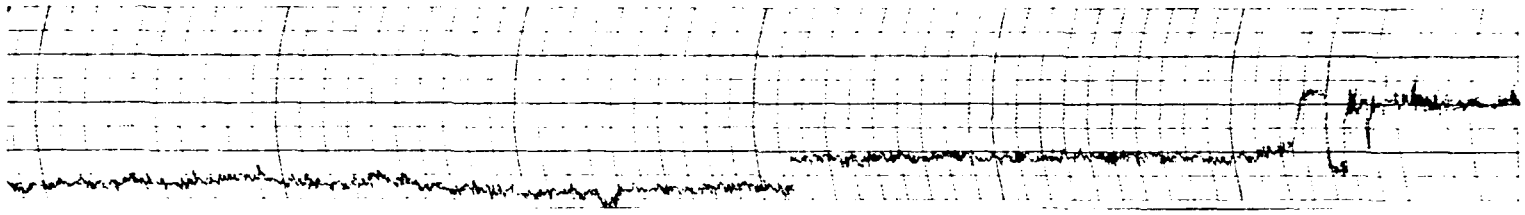


Column 1

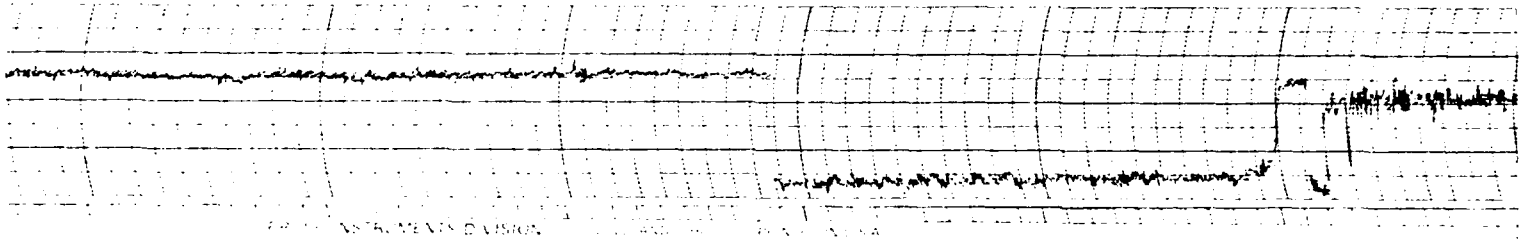


Column 4

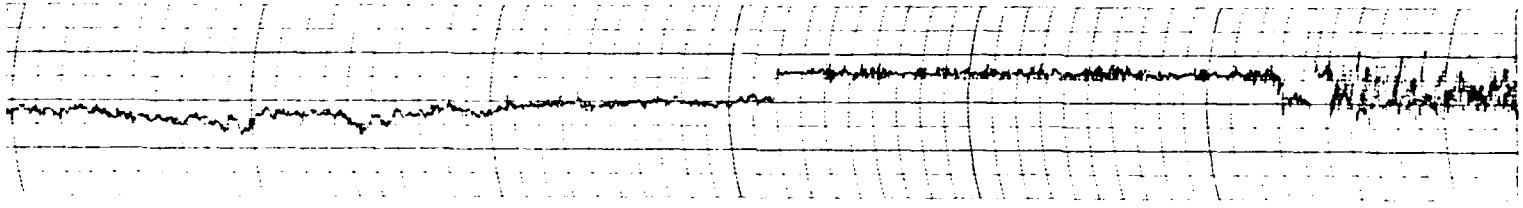
EEG Data, Day 3, Row 1, Differential Mode



Column 1



Column 2



Column 3

APPENDIX B

EEG Analog Recordings

The strip chart recordings illustrated in this appendix are demultiplexed data collected from a single electrode. They are representative of a "typical" analog EEG output. All data shown here were recorded at 25mm/sec (millimeters per second).

Date/Day#	Tape Counter Position #	Comments
Nov/8	1412	Gain set at 5000
	1435	Data starts
	1440	Start 60 count flash test, differential mode
	1467	Test ends
	1472	Start 60 count flash test, Absolute mode
	1495	Test ends
3 Nov/9	1508	Gain set at 2000
		No normal data recorded this day
	1528	Start 60 count flash test, absolute mode
	1561	Test ends
	1595	Preliminary differential mode test
	1627	Start differential mode test
	1631	Test ends
4 Nov/10	1644	Gain set at 50
		Gain to low for data to be useful
		Amplifier #5 used
	1809	60 count differential test begins
	1828	Test ends
5 Nov/11	1848	Amplifier #5 used
	1866	60 count differential test begins
	1885	Test ends
	1920	60 count absolute test begins
	1938	Test ends

APPENDIX A

Data Tape Log

Date/Day#	Tape Counter Position #	Comments
27 Oct/3	109	Gain set at 500
	114	Ricky stimulated w/strobe
	198	End of Data
28 Oct/4	232	Gain set at 1000
	275	Amplifiers on, testing begins normal mod
	290	Strobe starts, VER data
	306	100 Count test begins
	332	Test ends
29 Oct/5	528	Voice track starts
	550	Data starts
	570	100 count VER test begins
	624	Test ends
	626	Normal test starts
	670	Test ends
30 Oct/6	819	100 count normal test begins
	836	test ends
	838	100 count EVR test begins
	907	Test ends
	1000	Test with bucket over Ricky's head
	1065	Test with bucket over strobe
Nov/7	1140	Gain set at 1000 New strobe is introduced
	1285	Start 60 count flash test, differential mode
	1308	Test ends
	1315	Start 60 count flash test, absolute mode
	1349	Test ends
	1360	Start 60 count flash test, differential mode, strobe covered
	1385	Test ends

2. The spiked strobe should be used from the outset.
3. Good documentation should be kept throughout the experiment.
4. The spatio-temporal correlation technique should be accomplished.
5. Other signal processing techniques, such as Kalman filtering and Fourier and Walsh transformations, could be attempted and their results compared.

guous file space for data collection added some more problems. Although contiguous file data collection means faster collection; on a system with no repack capability for the hard disc, this means limited file lengths. A work around is to collect directly onto a floppy disc and then the file can be demultiplexed into the column outputs and these files also stored on floppy disc. Even then, memory size limits collection to under 300,000 samples at a time. This system turned out to be adequate for this thesis. However, the follow on thesis using the 256 electrode array will definitely need a much larger contiguous file spce and memory capacity since sixteen channels will be sampled instead of five.

The stored data and the Ampex machine, which was used to play the data back, brought in another set of problems. First, the data were well infused with sixty cycle hum when it was recorded. This is very apparent when looking at some of the DEC strip charts. Also, at some point the data became distorted with noise and jitter. The tape recorder would also develop squeaks along certain portions of the tape which also caused tremendous distortion in the data. Many times the data had to be collected over again in the hopes that this squeak would not appear. Fortunately this recorder need not be used in further implant experiments.

Recommendations for further implant experiments include:

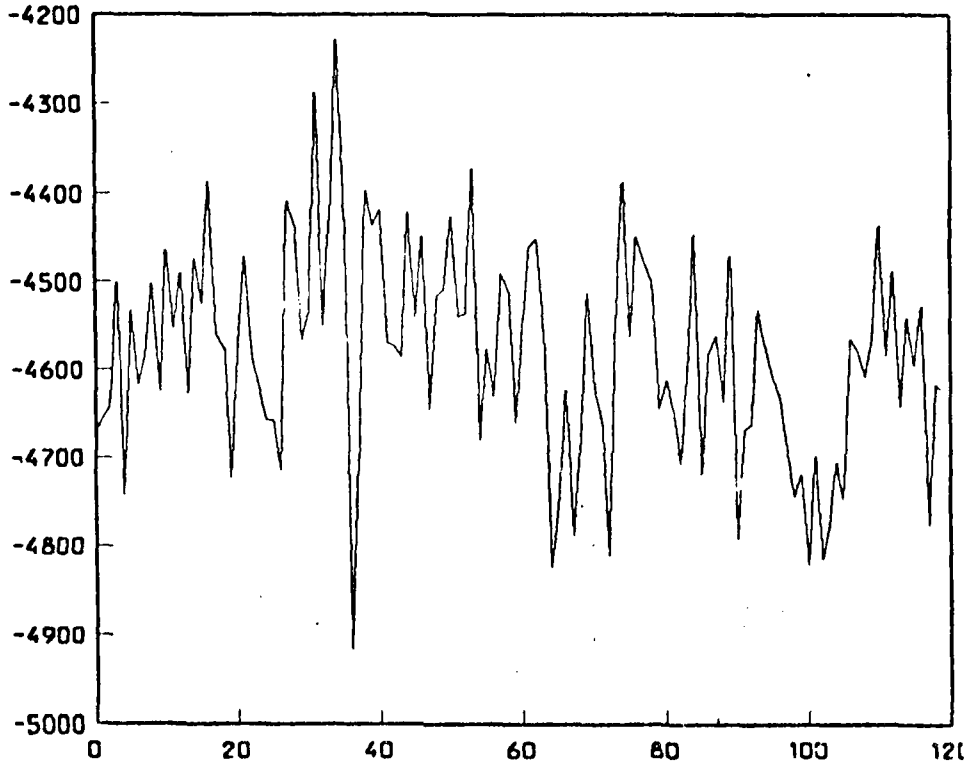
1. Time should be spent preparing a plan for testing with control tests included for comparison purposes.

efforts were negligent. Their problem was lack of time. For example, Hensley and Denton collected their data in late October thru early November, a time when their writing should have been nearly complete. For future experiments, a plan of analysis should be written down prior and followed as closely as possible with a written log kept for future reference.

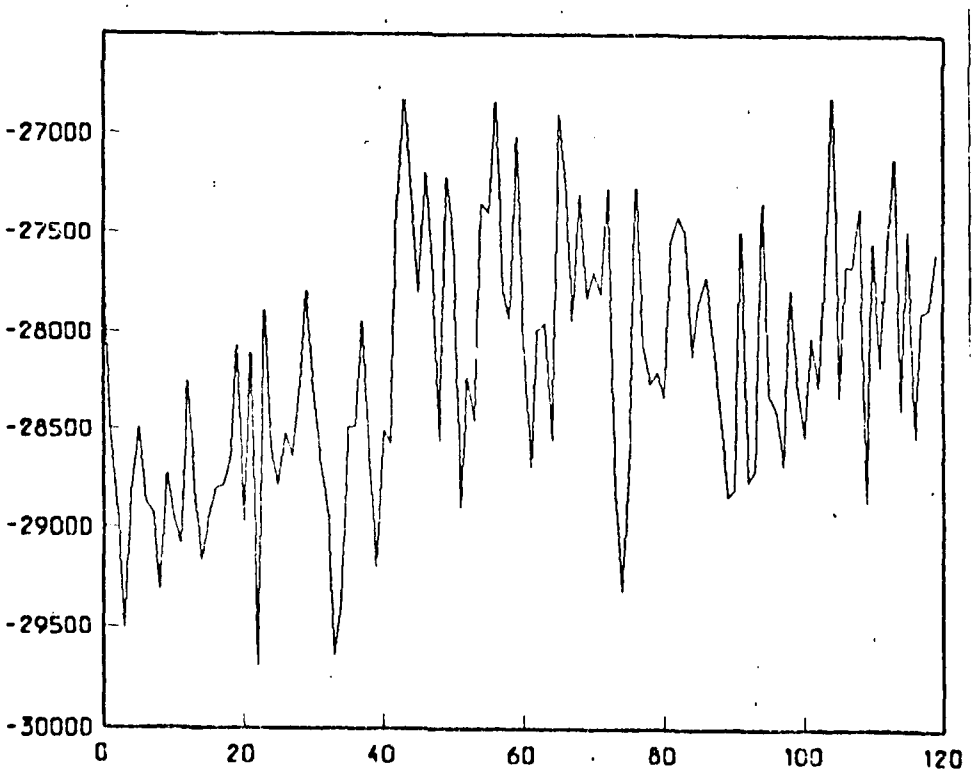
Another area of major problems lay with the computer. Poor documentation caused many of these problems. The learning curve for this computer turned out to be tremendous. Anyone who plans to attempt further data acquisition and analysis with the MASSCOMP MC-500 should at least attend the data acquisition course given by Masscomp in Boston. It would also be very helpful to attend the graphics course, if time and money are available. A background in computers and computer programming (especially "C") would be invaluable.

Lack of needed equipment caused more problems for which some work-arounds had to be devised. A D/A board was needed to obtain the strip chart recordings of the EEG signals from the individual electrodes. This board had to be borrowed from another lab on base whenever it was needed. Also, another one of the thesis objectives could not be accomplished because neither the magnetic tape drive nor a modem had come in (both had been ordered). The tape drive should be available for the next experiment. It is highly recommended that the spatio-temporal correlation technique be tested with the larger array's data.

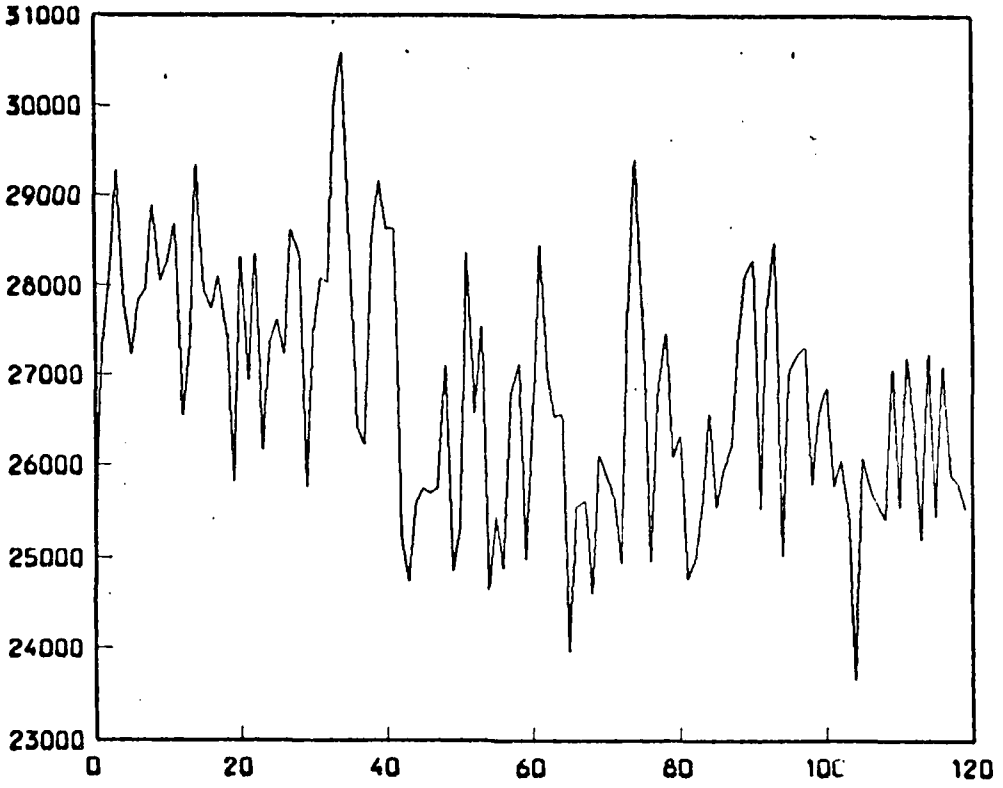
The fact that the data acquisition programs need conti-



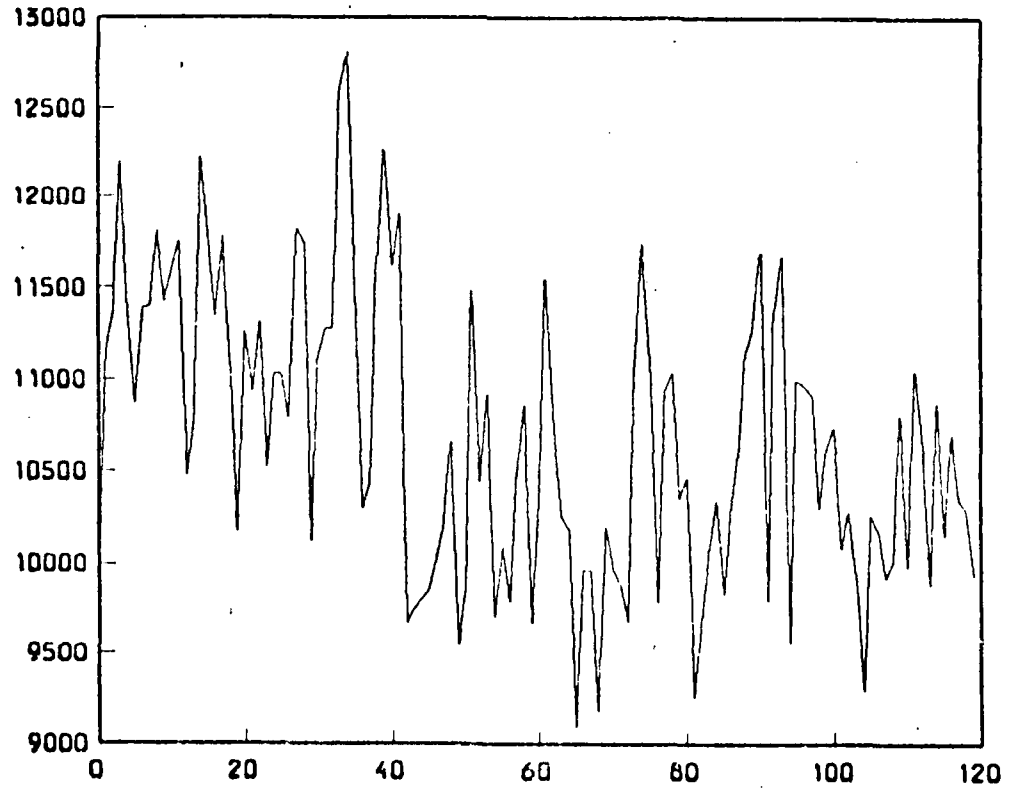
DAY 3, DIFFERENTIAL TEST, ROW 1, COLUMN 1



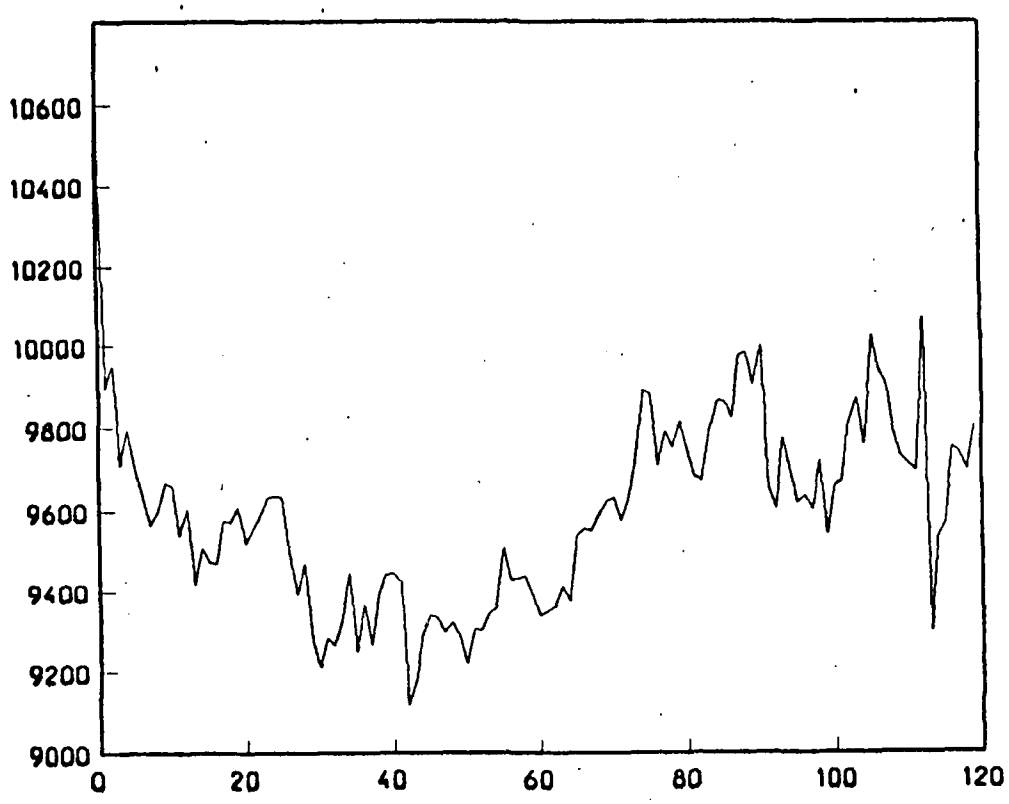
DAY 3, DIFFERENTIAL TEST, ROW 1, COLUMN 2



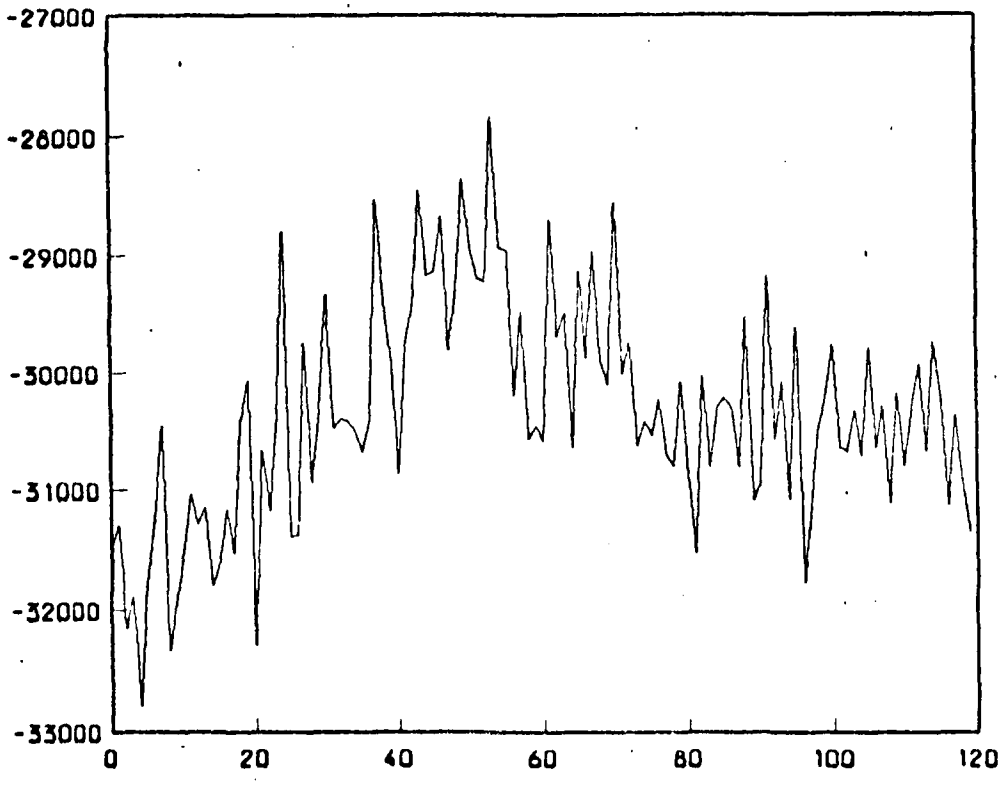
DAY 3, DIFFERENTIAL TEST, ROW 1, COLUMN 3



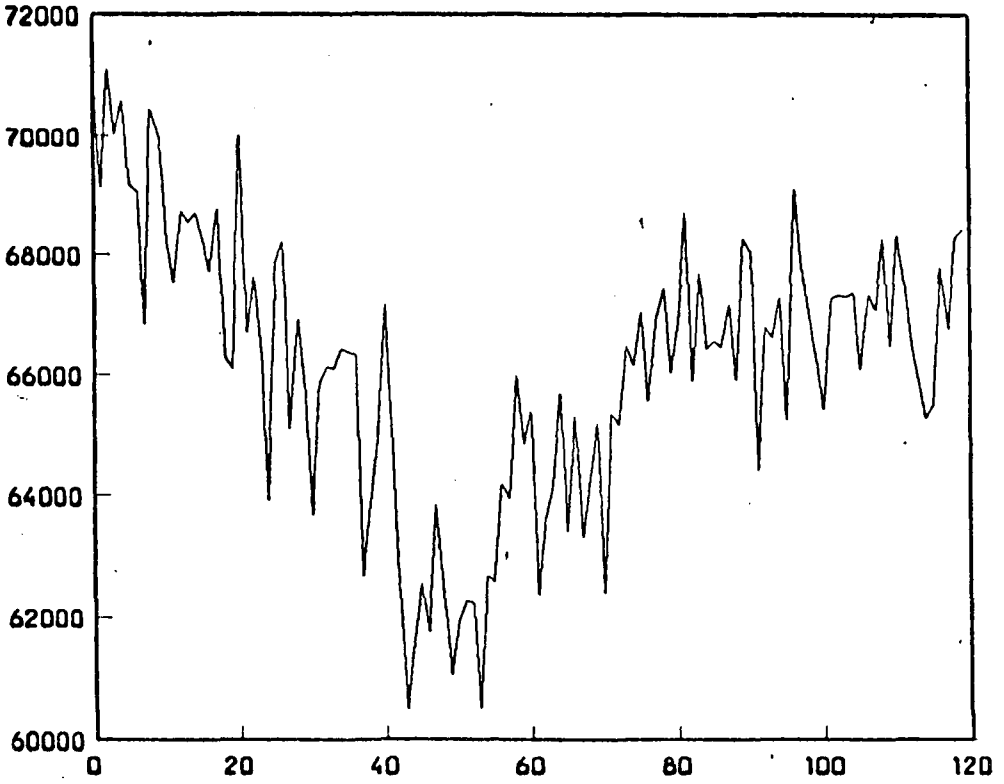
DAY 3, DIFFERENTIAL TEST, ROW 1, COLUMN 4



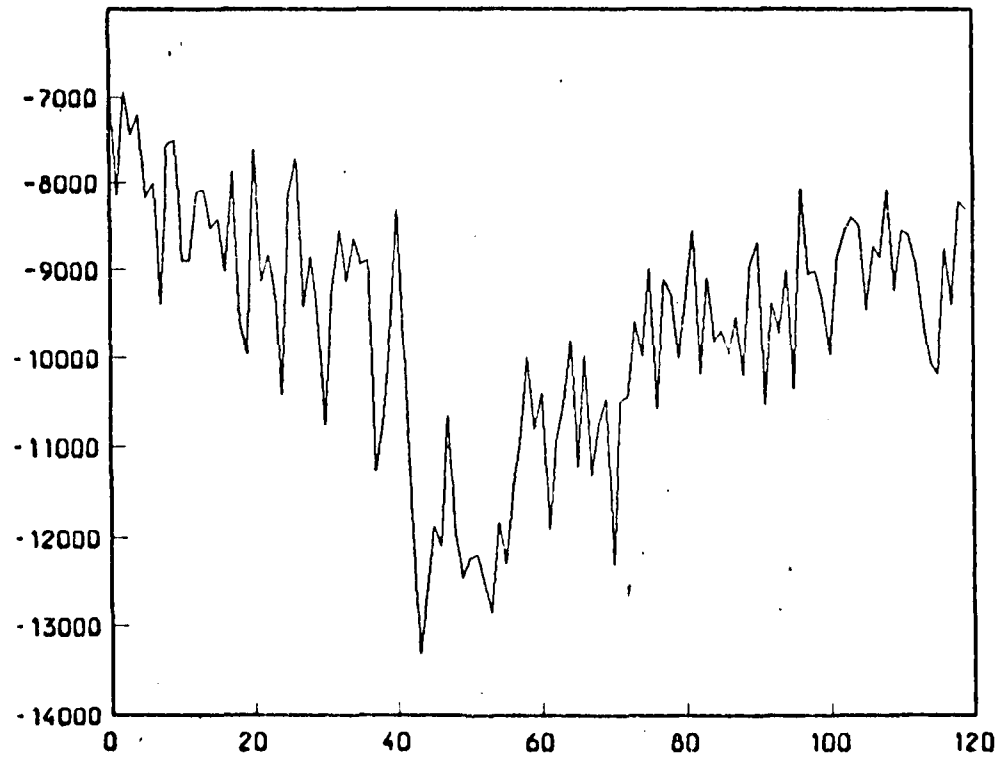
DAY 3, DIFFERENTIAL TEST, ROW 2, COLUMN 1



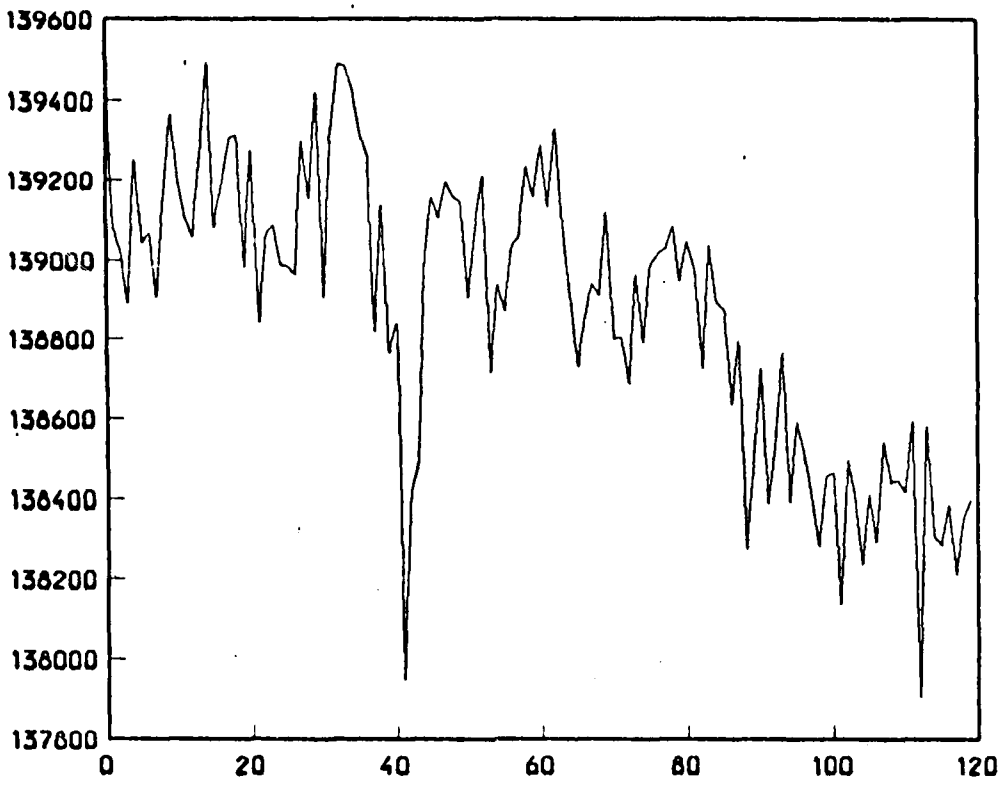
DAY 3, DIFFERENTIAL TEST, ROW 2, COLUMN 2



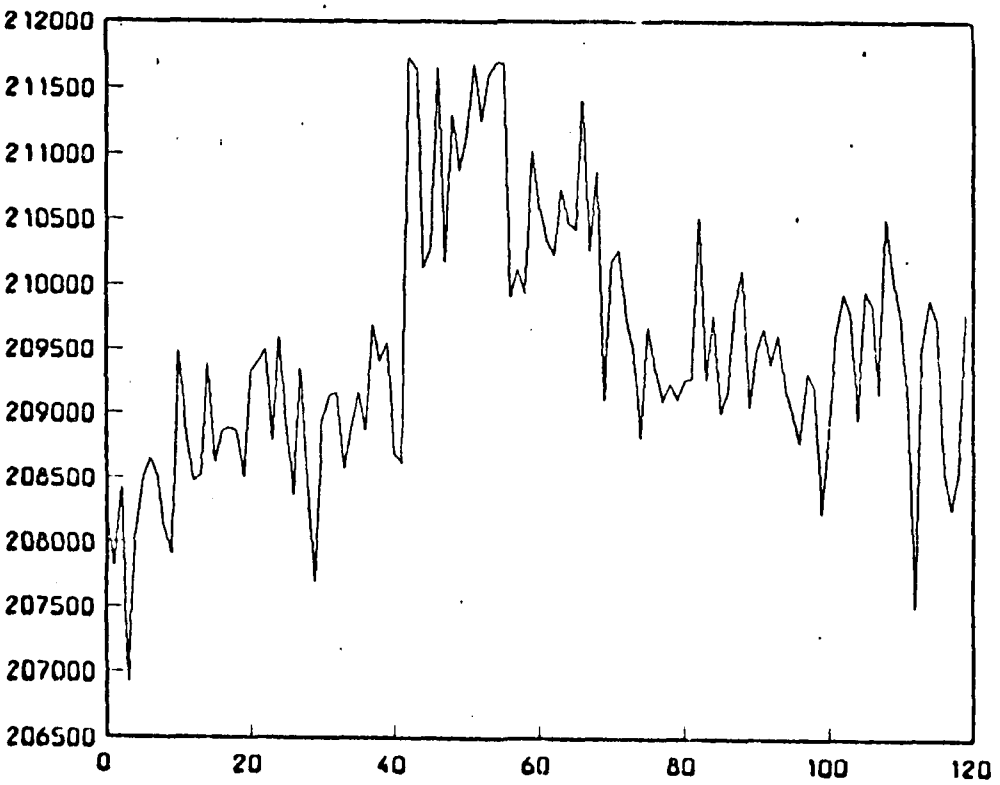
DAY 3, DIFFERENTIAL TEST, ROW 2, COLUMN 3



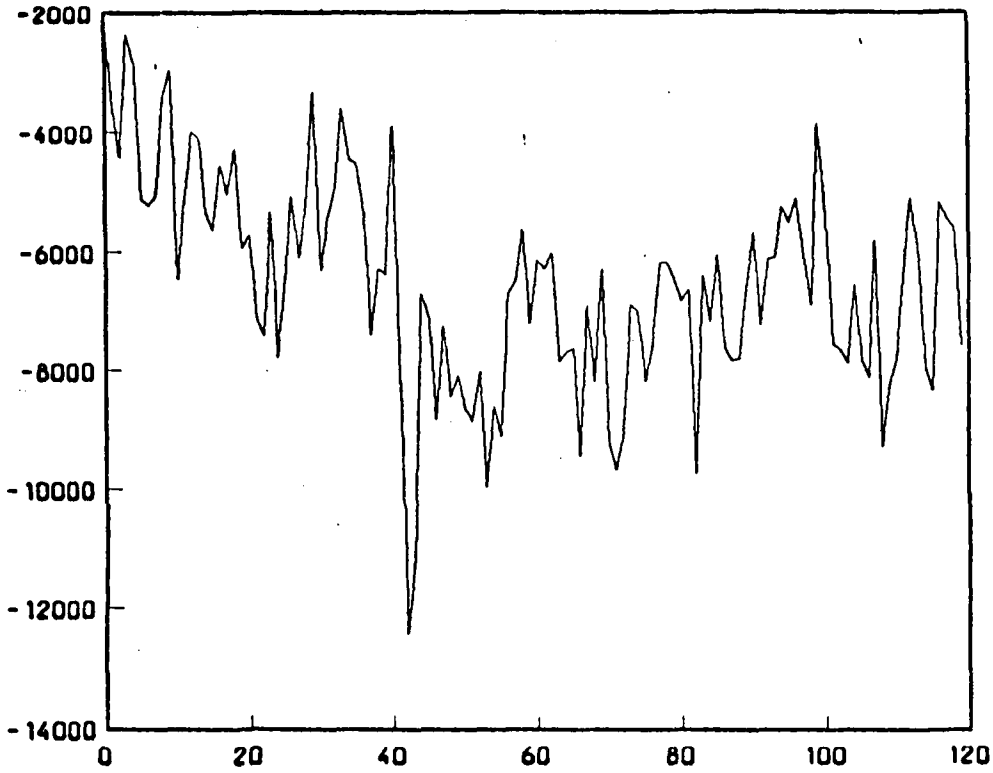
DAY 3, DIFFERENTIAL TEST, ROW 2, COLUMN 4



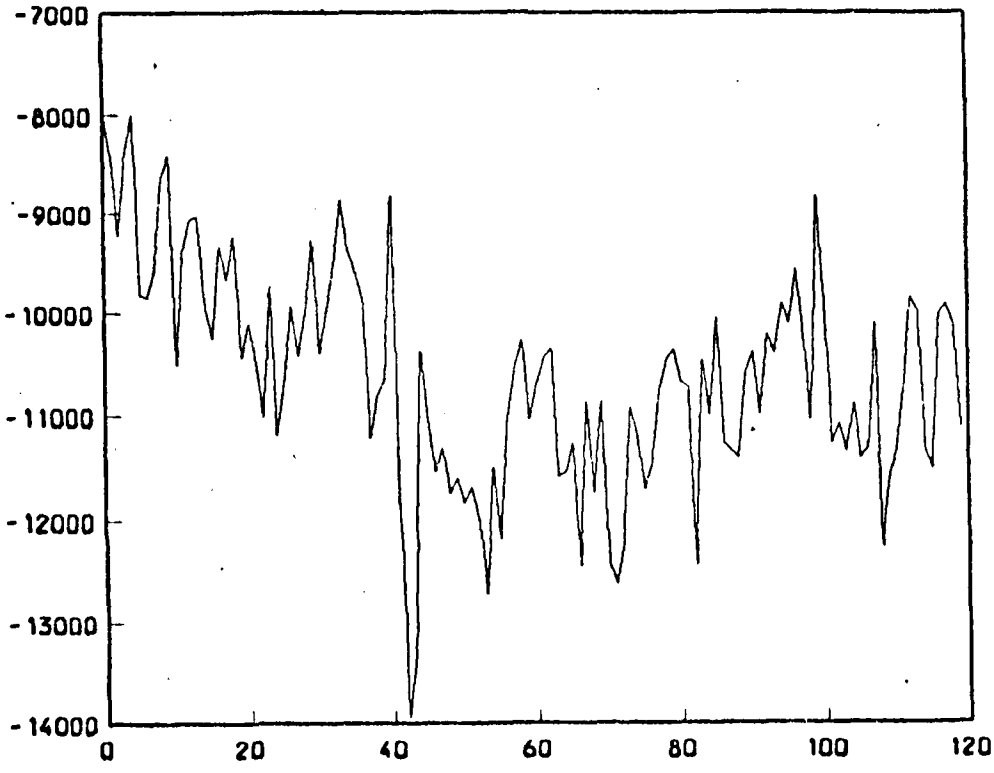
DAY 3, DIFFERENTIAL TEST, ROW 3, COLUMN 1



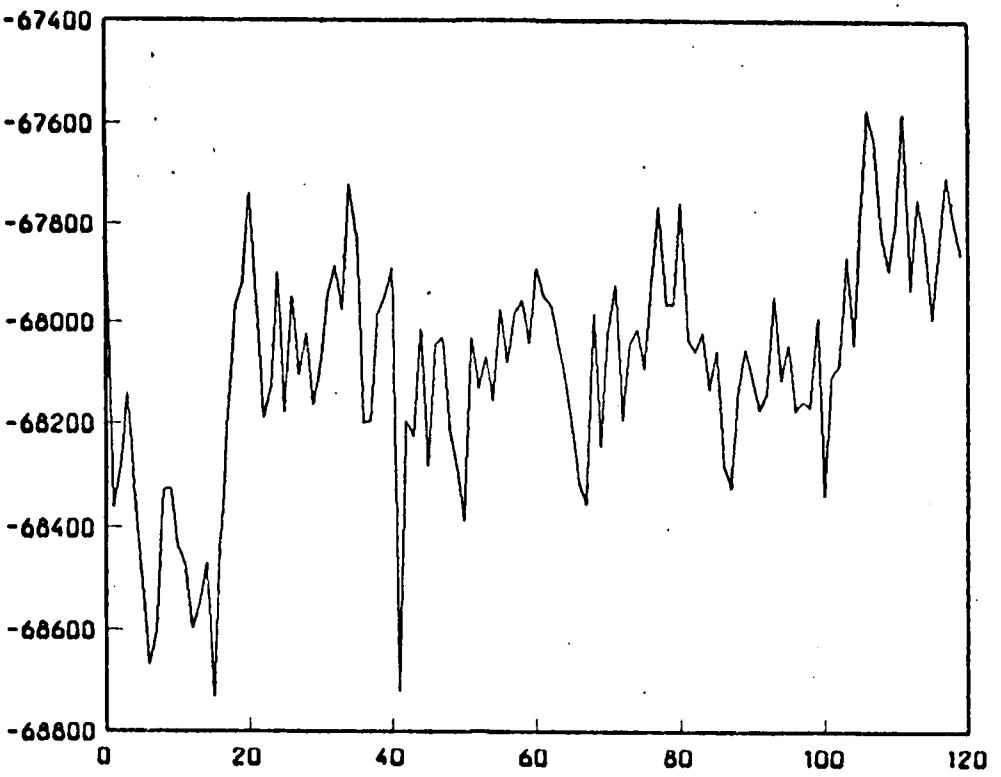
DAY 3, DIFFERENTIAL TEST, ROW 3, COLUMN 2



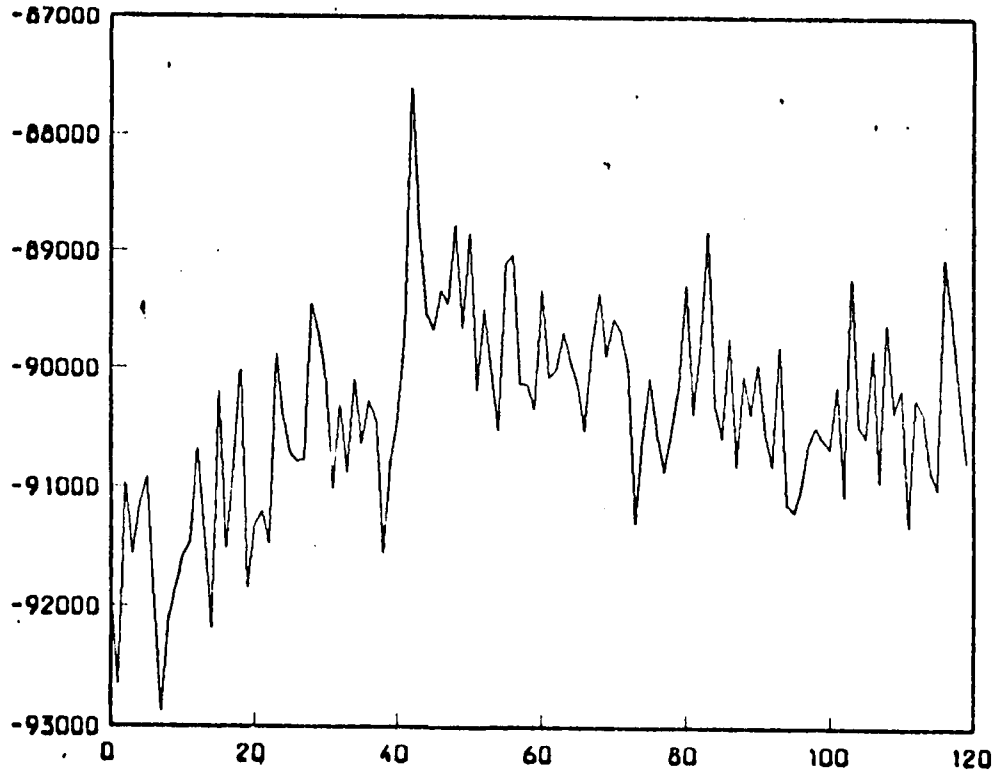
DAY 3, DIFFERENTIAL TEST, ROW 3, COLUMN 3



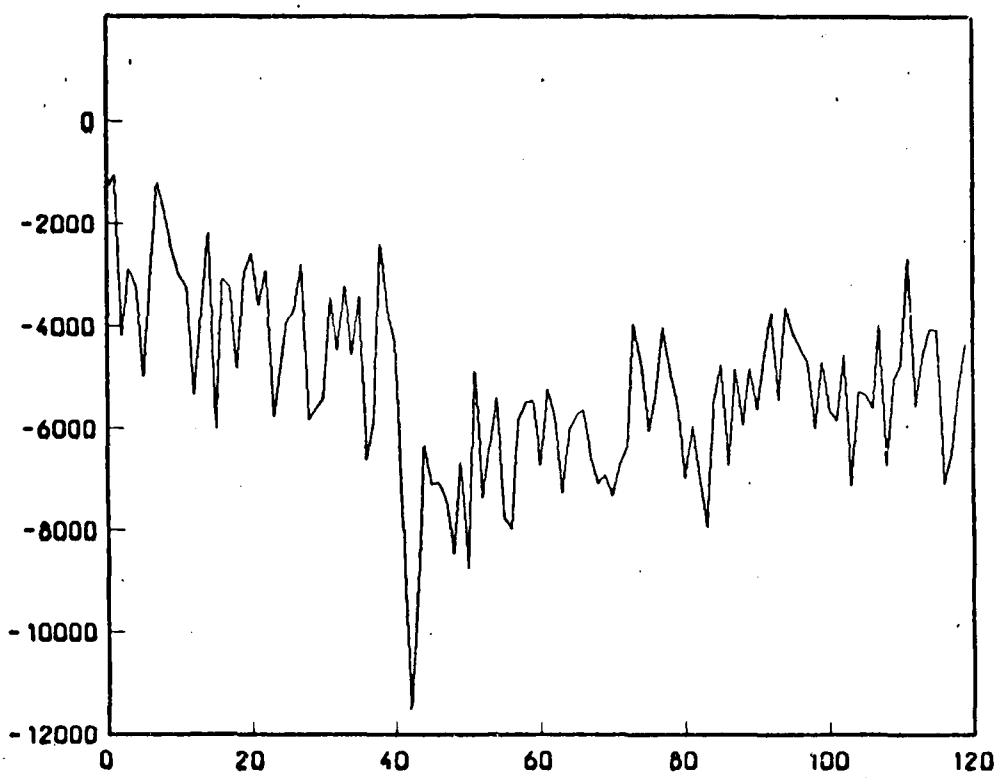
DAY 3, DIFFERENTIAL TEST, ROW 3, COLUMN 4



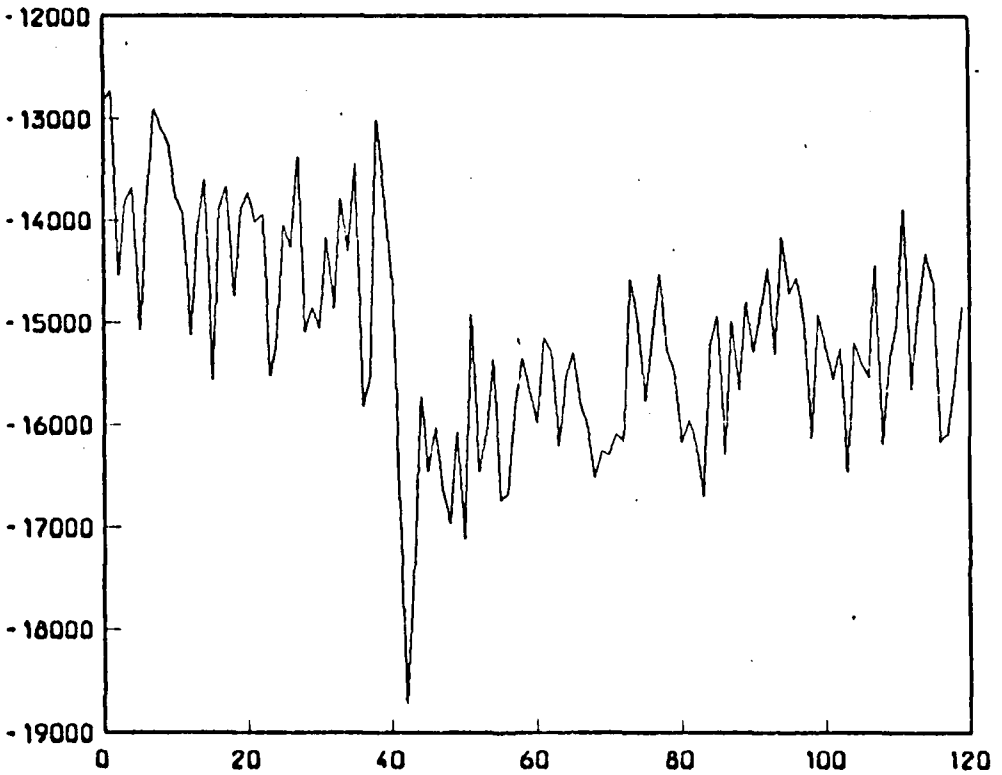
DAY 3, DIFFERENTIAL TEST, ROW 4, COLUMN 1



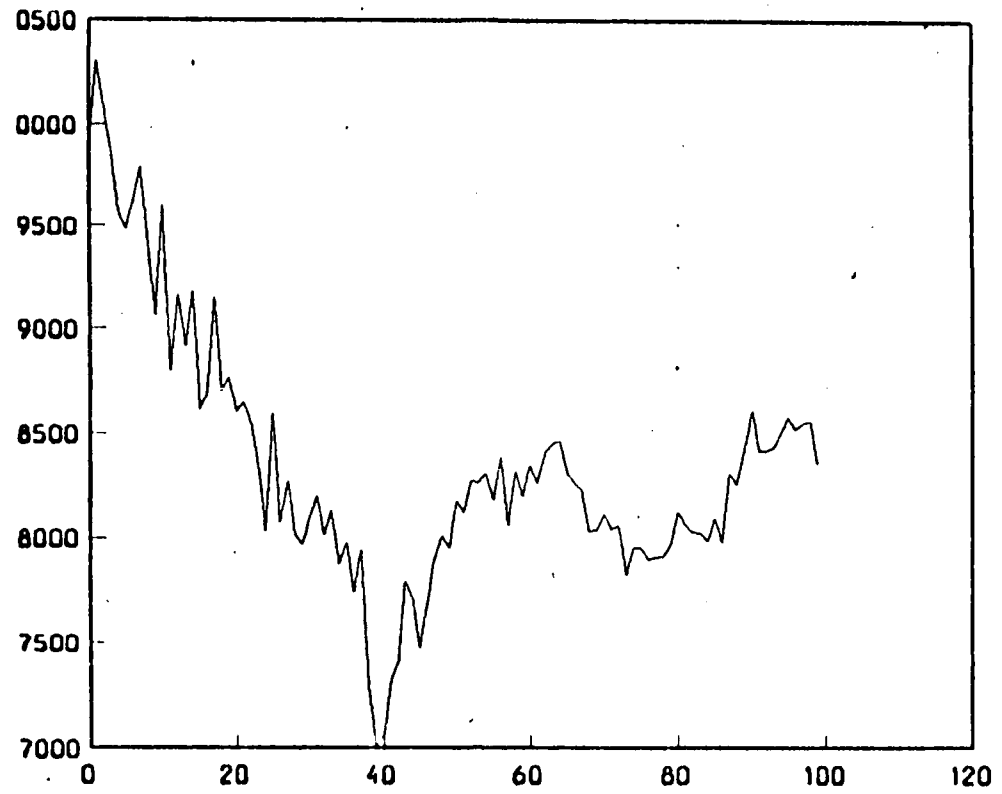
DAY 3, DIFFERENTIAL TEST, ROW 4, COLUMN 2



DAY 3, DIFFERENTIAL TEST, ROW 4, COLUMN 3

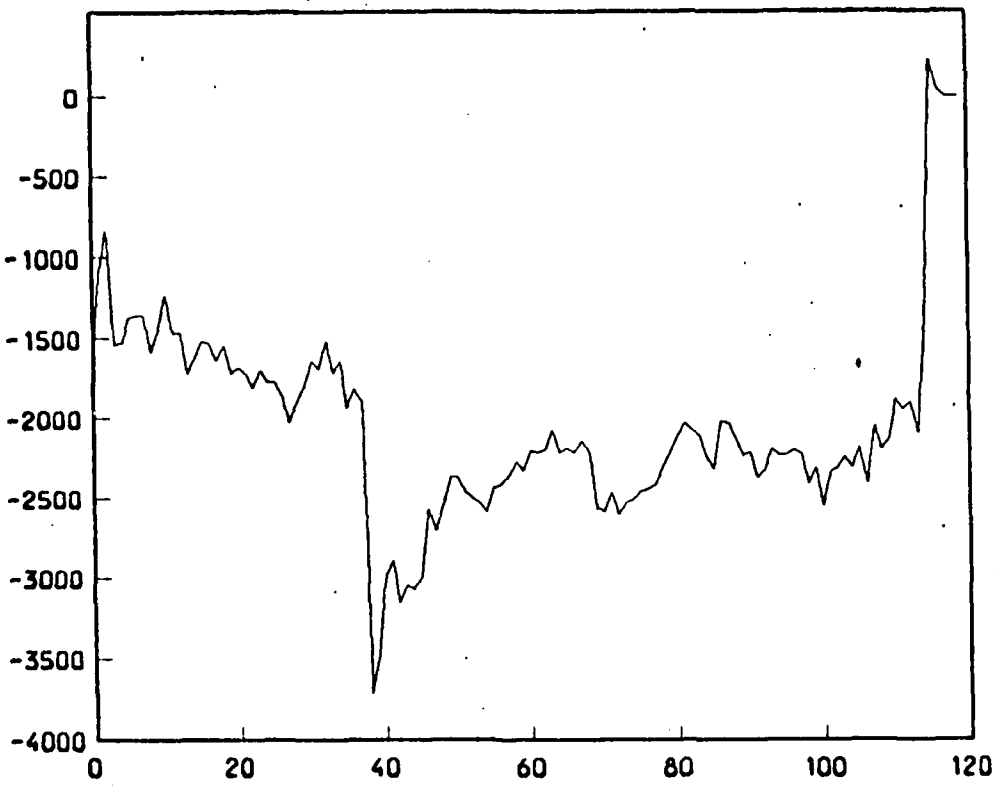


DAY 3, DIFFERENTIAL TEST, ROW 4, COLUMN 4

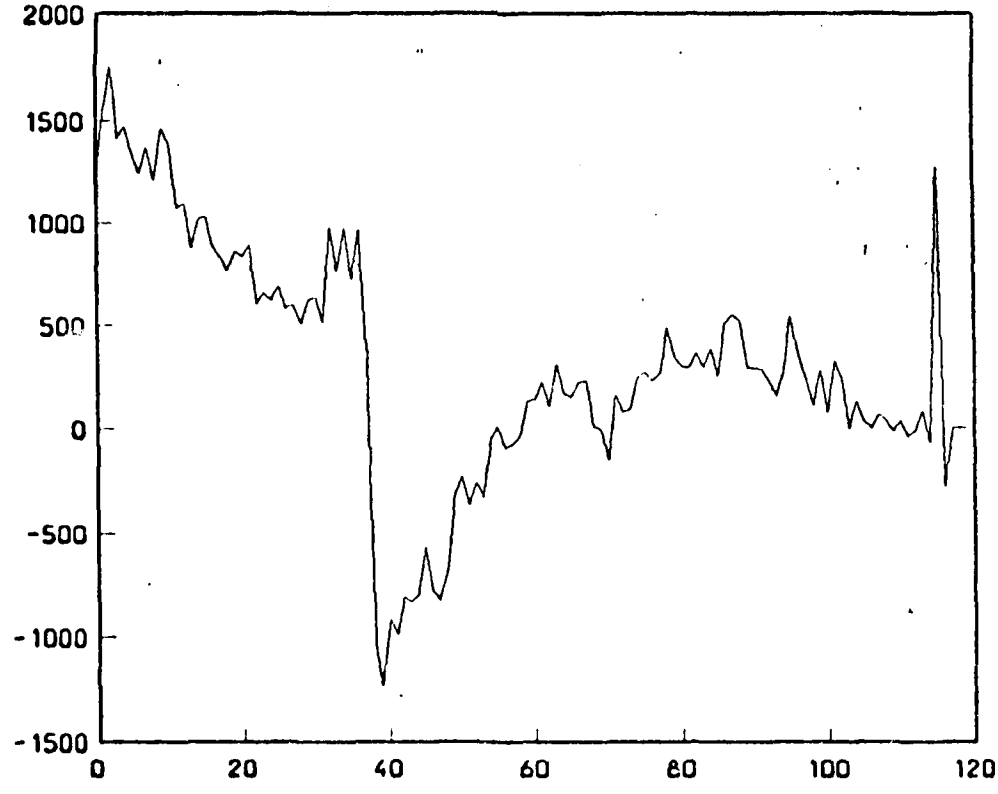


DAY 4, DIFFERENTIAL TEST, ROW 1, COLUMN 1

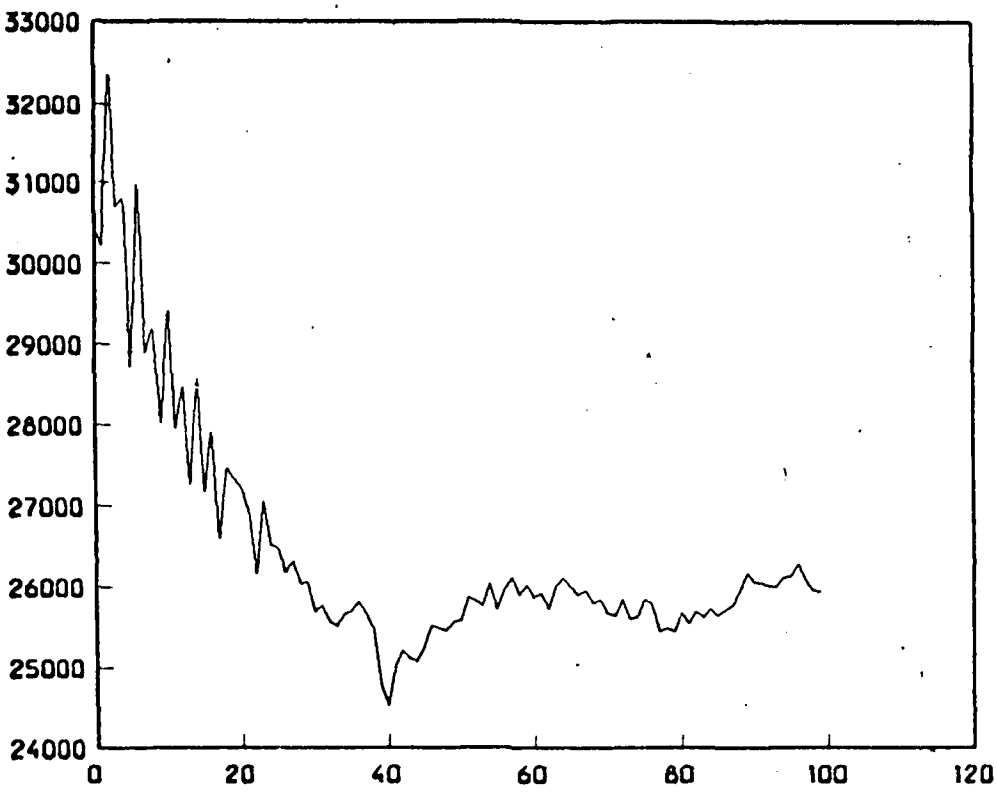
DAY 4, DIFFERENTIAL TEST, ROW 1, COLUMN 2



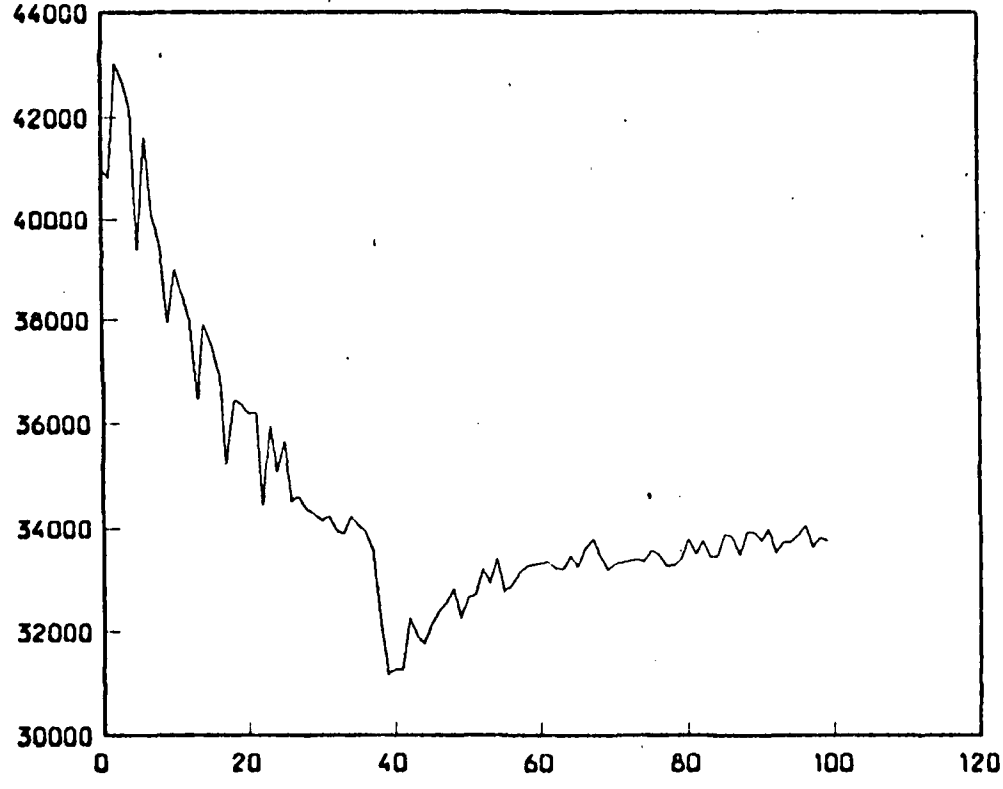
DAY 4, DIFFERENTIAL TEST, ROW 1, COLUMN 3



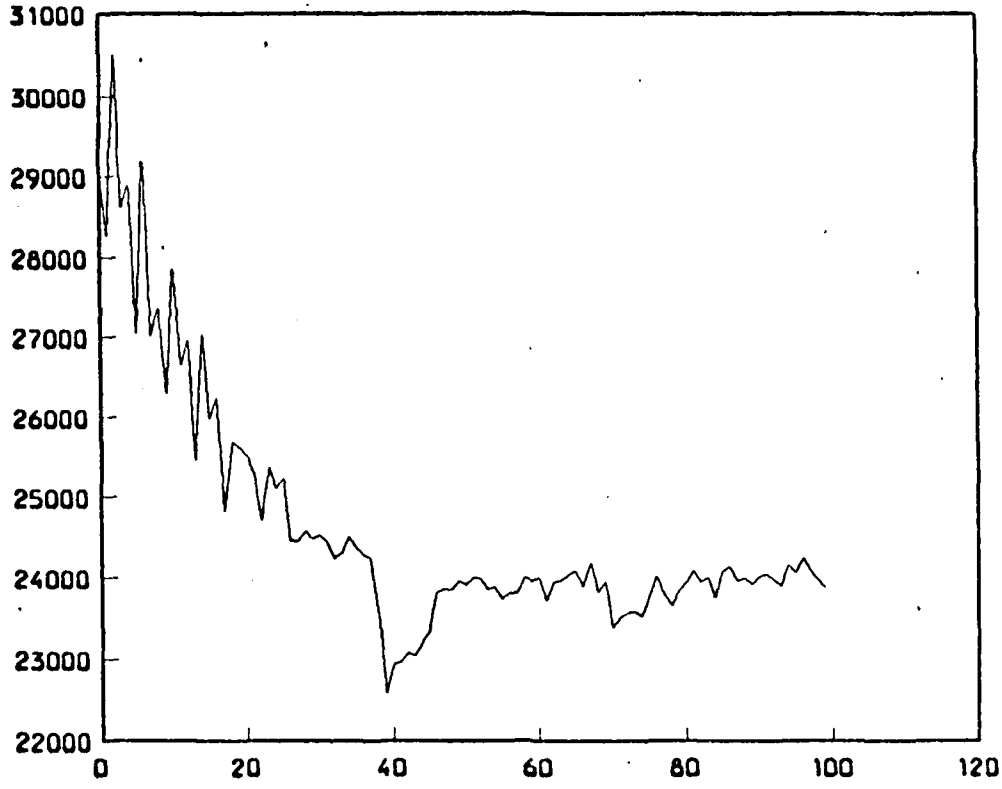
DAY 4, DIFFERENTIAL TEST, ROW 1, COLUMN 4



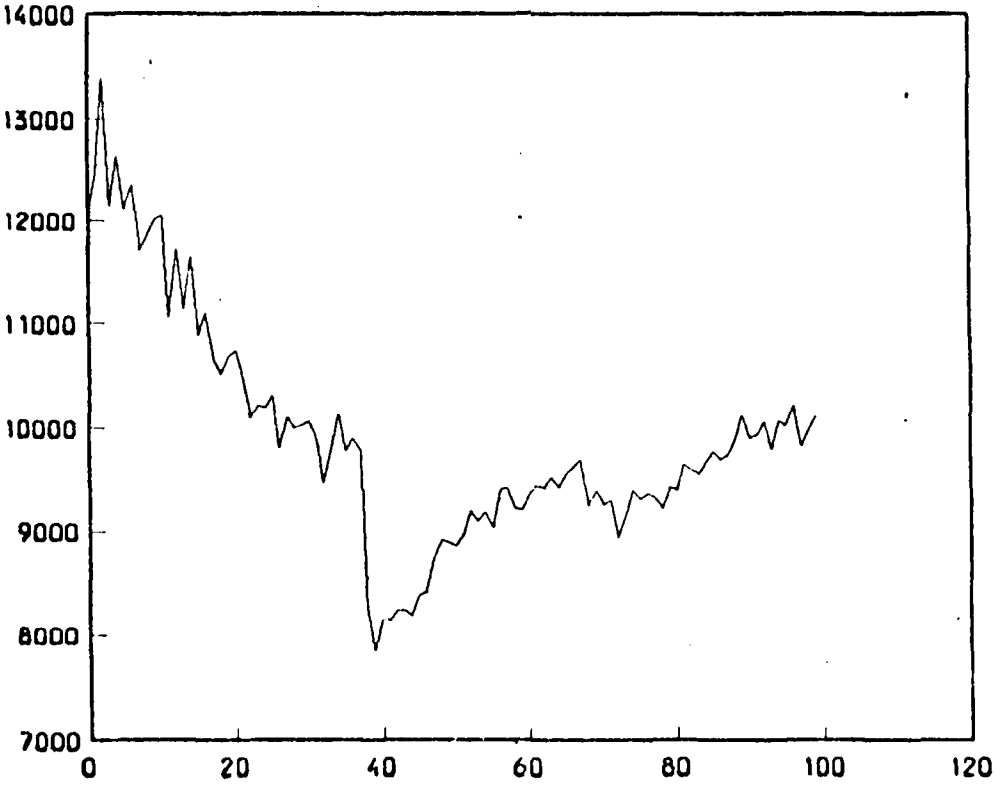
DAY 4, DIFFERENTIAL TEST, ROW 2, COLUMN 1



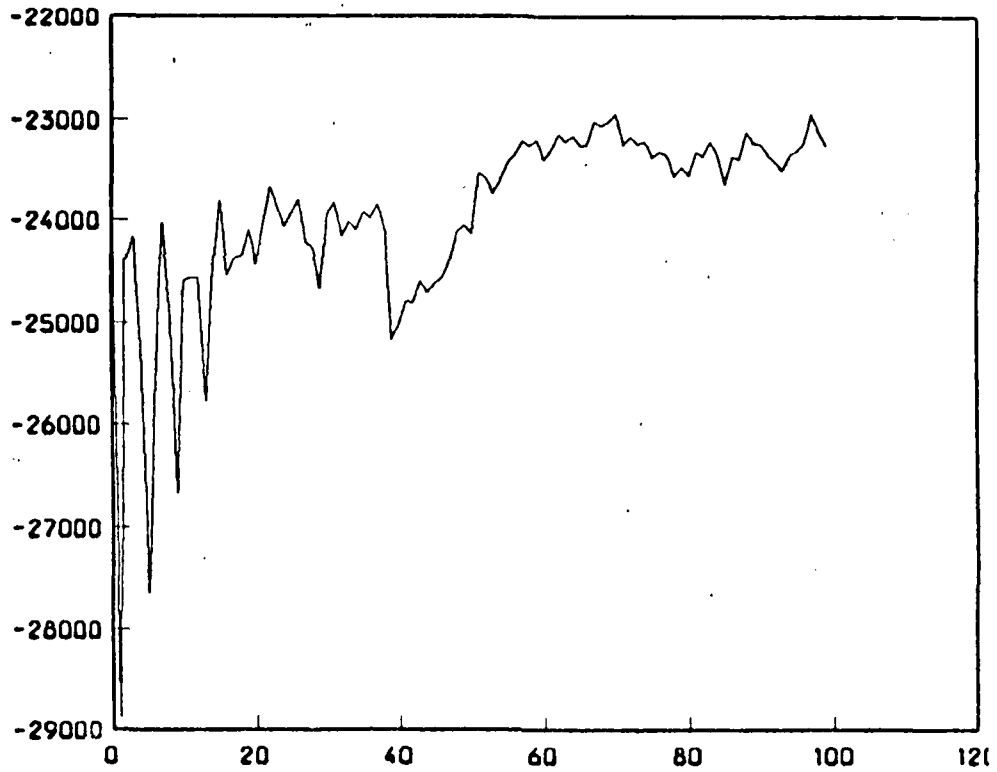
DAY 4, DIFFERENTIAL TEST, ROW 2, COLUMN 2



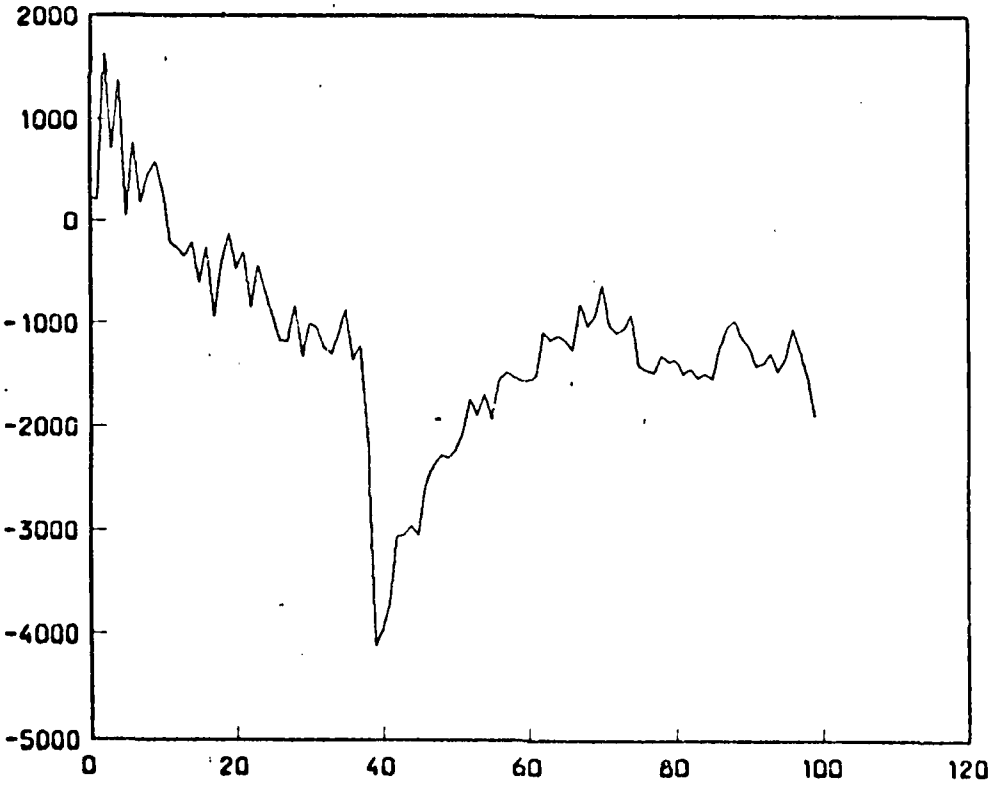
DAY 4, DIFFERENTIAL TEST, ROW 2, COLUMN 3



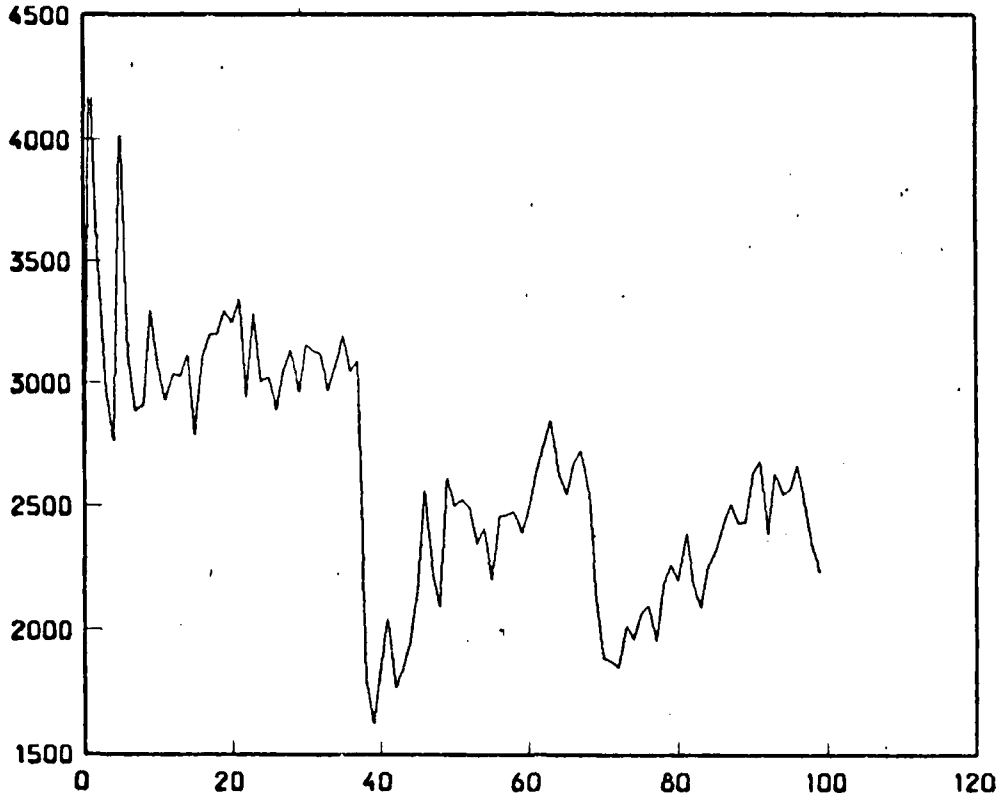
DAY 4, DIFFERENTIAL TEST, ROW 2, COLUMN 4



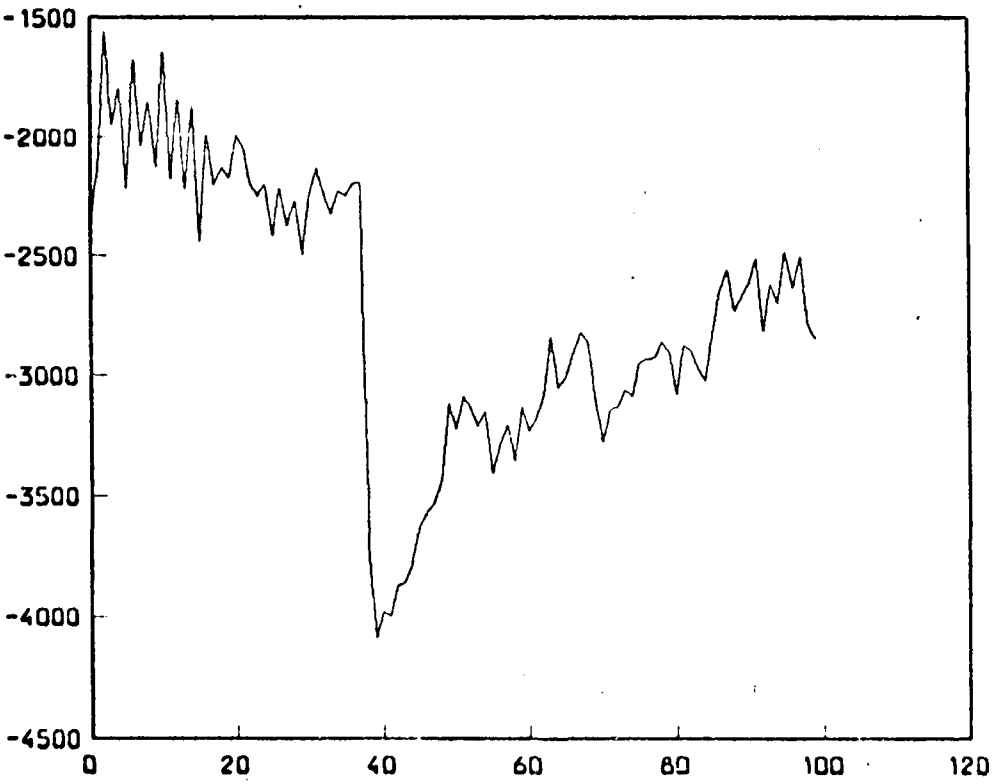
DAY 4, DIFFERENTIAL TEST, ROW 3, COLUMN 1



DAY 4, DIFFERENTIAL TEST, ROW 3, COLUMN 2



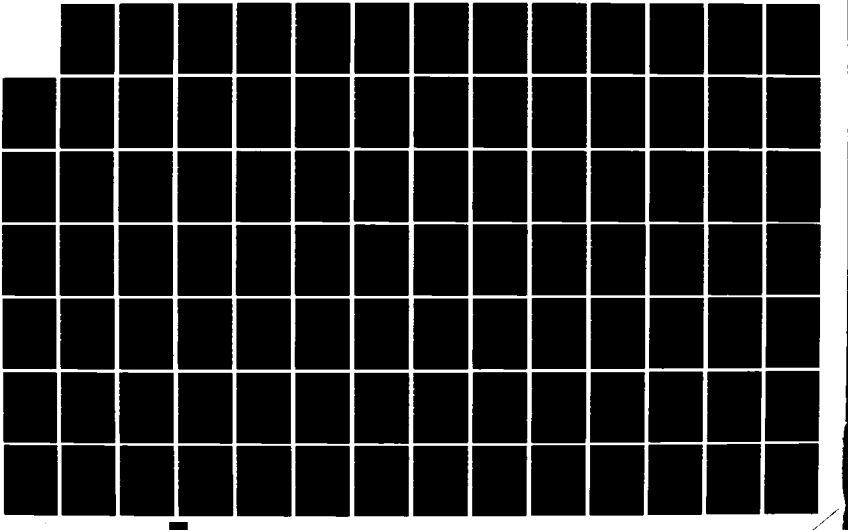
DAY 4, DIFFERENTIAL TEST, ROW 3, COLUMN 3

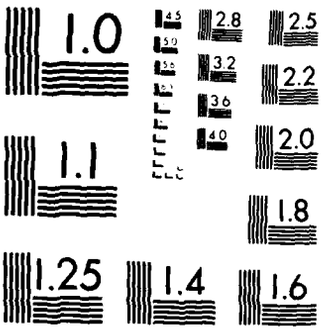


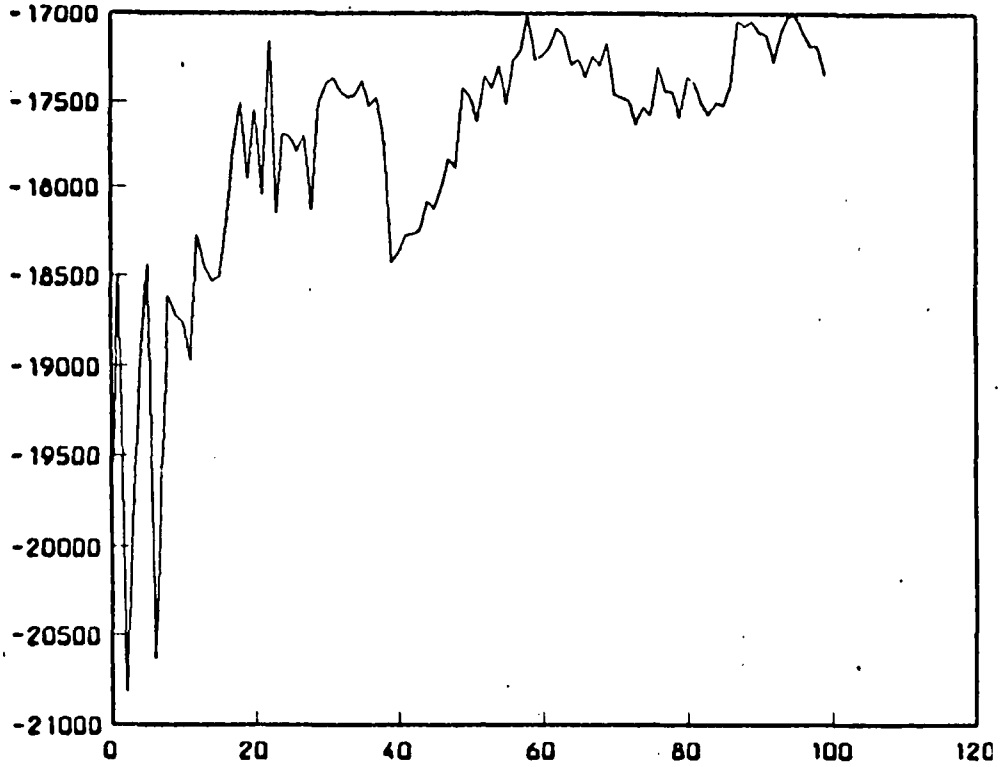
DAY 4, DIFFERENTIAL TEST, ROW 3, COLUMN 4

AD-A151 699 THE FIRST CORTICAL IMPLANT OF A SEMICONDUCTOR MULTIELECTRODE ARRAY: CONTI.. (U) AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOOL OF ENGI.. 2/3

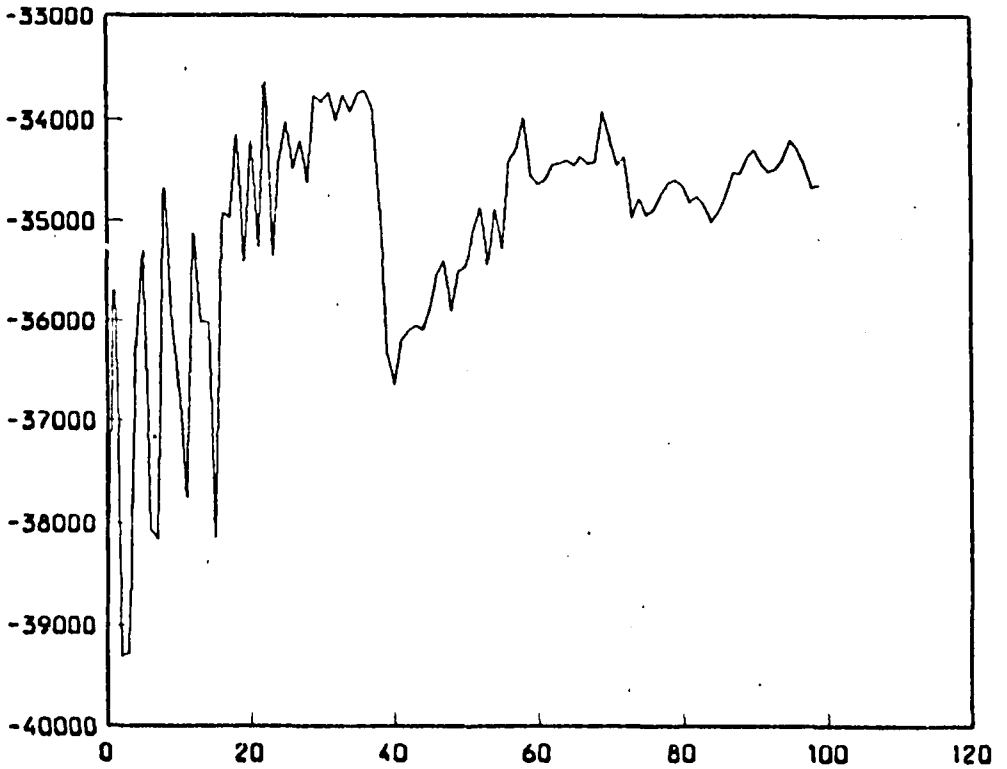
UNCLASSIFIED J D COLLIER ET AL. DEC 84 F/G 5/10 NL



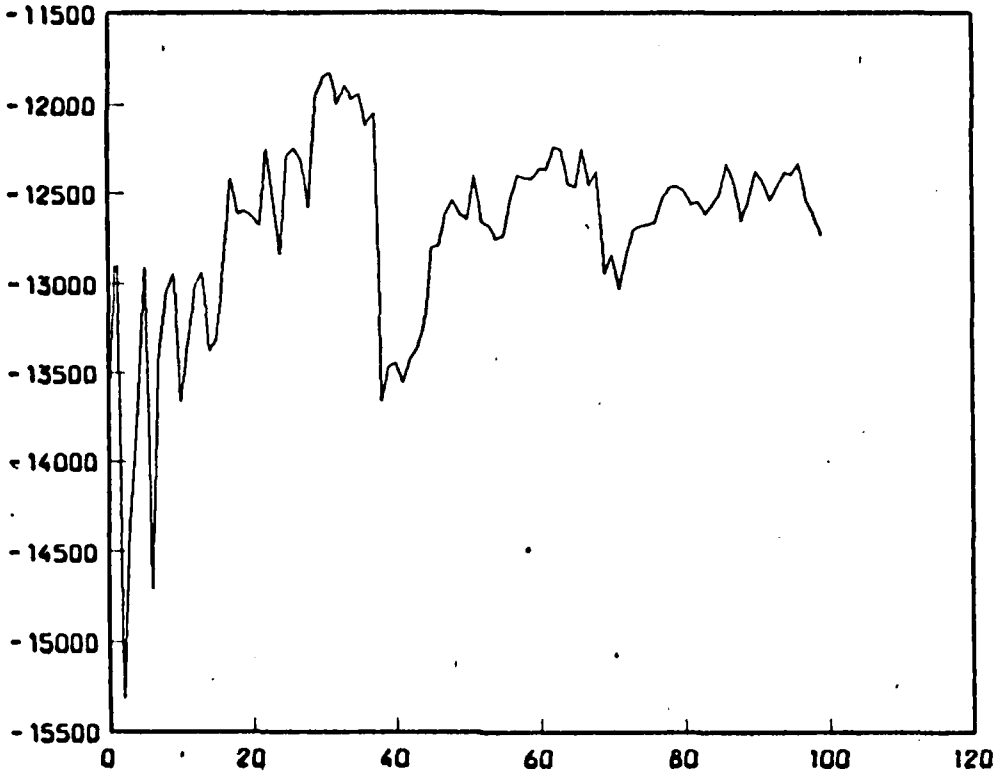




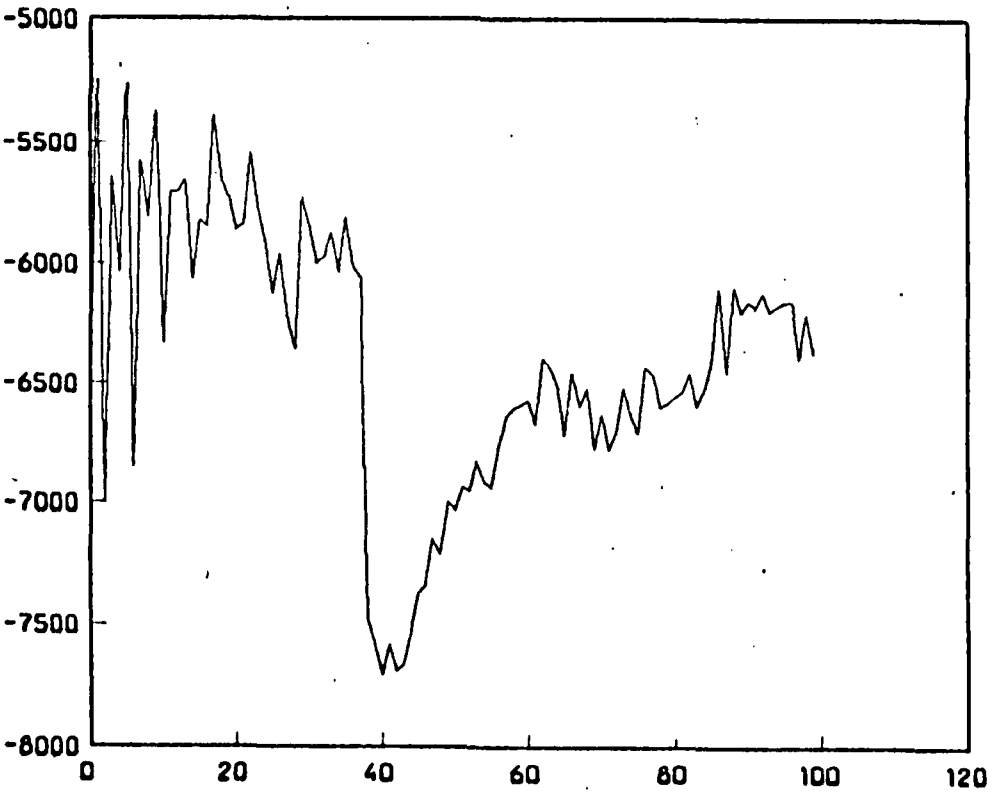
DAY 4, DIFFERENTIAL TEST, ROW 4, COLUMN 1



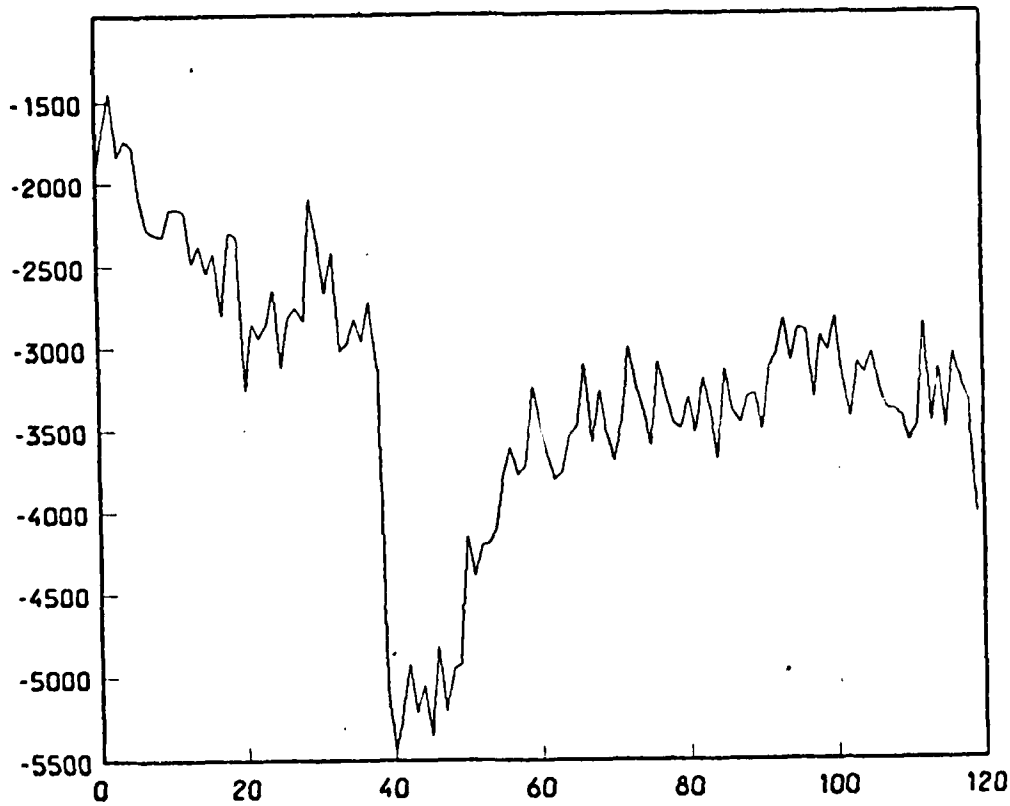
DAY 4, DIFFERENTIAL TEST, ROW 4, COLUMN 2



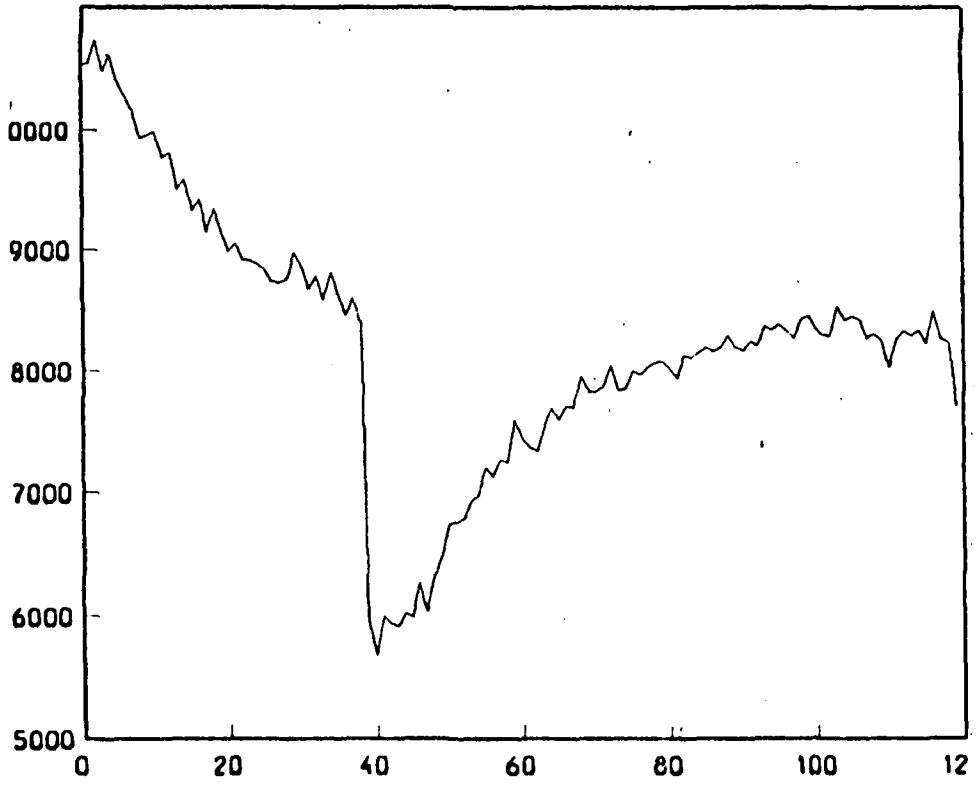
DAY 4, DIFFERENTIAL TEST, ROW 4, COLUMN 3



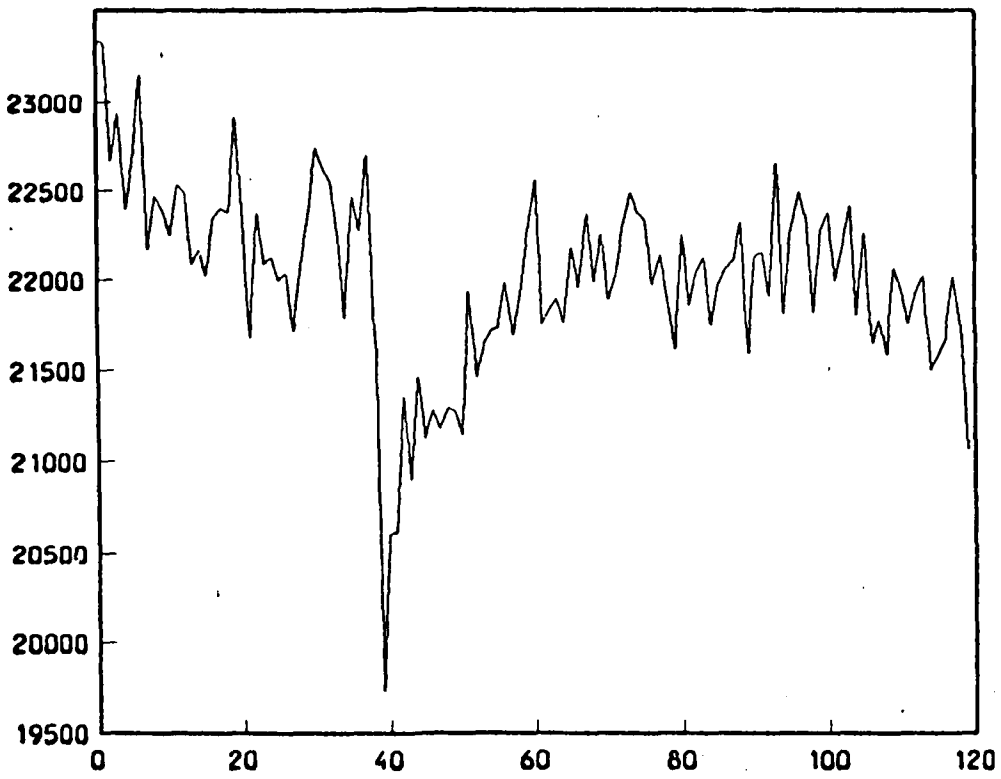
DAY 4, DIFFERENTIAL TEST, ROW 4, COLUMN 4



DAY 5, DIFFERENTIAL TEST, ROW 1, COLUMN 3



DAY 5, DIFFERENTIAL TEST, ROW 1, COLUMN 4

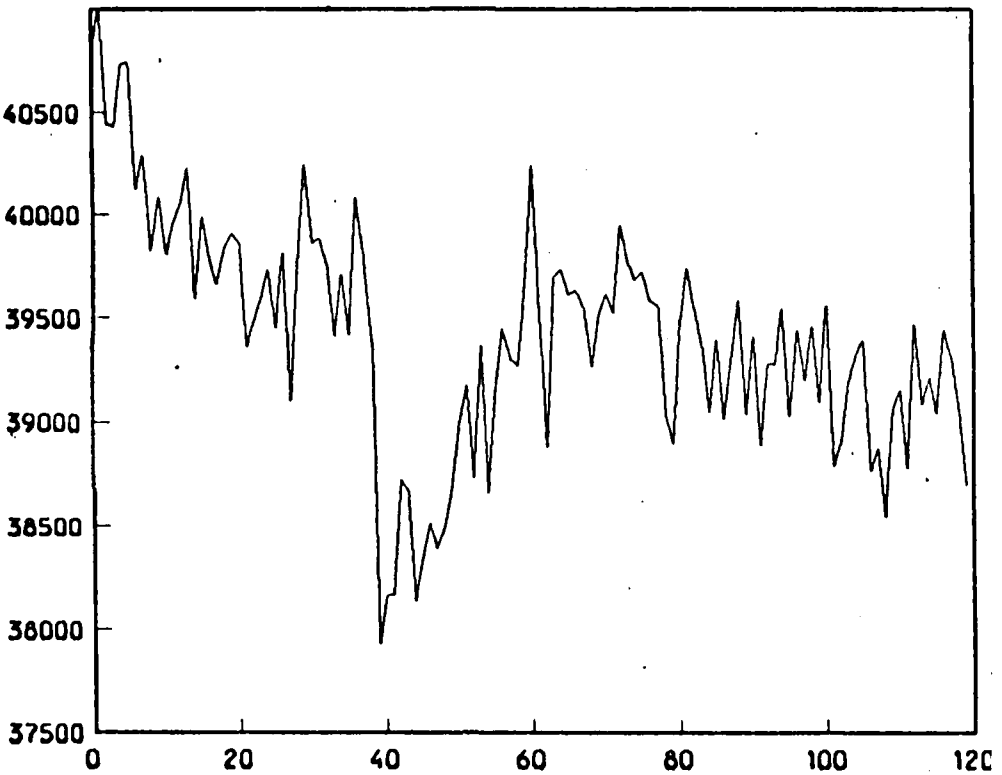


DAY 5, DIFFERENTIAL TEST, ROW 2, COLUMN 1

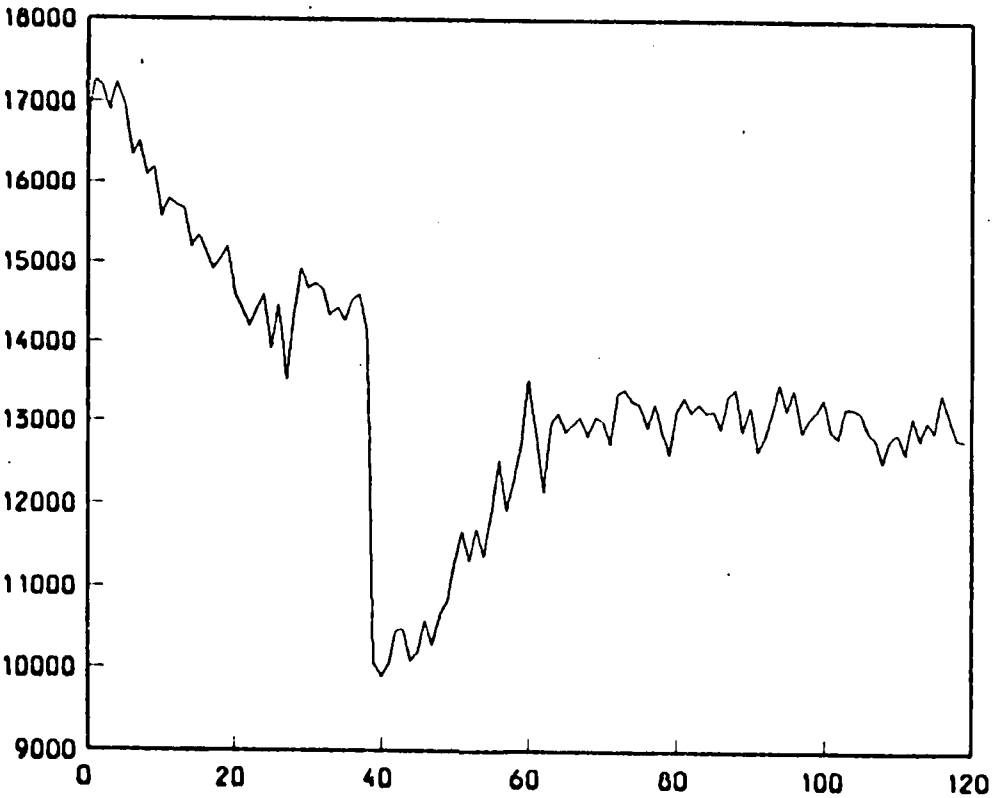
DAY 5, DIFFERENTIAL TEST, ROW 2, COLUMN 2

DAY 5, DIFFERENTIAL TEST, ROW 2, COLUMN 3

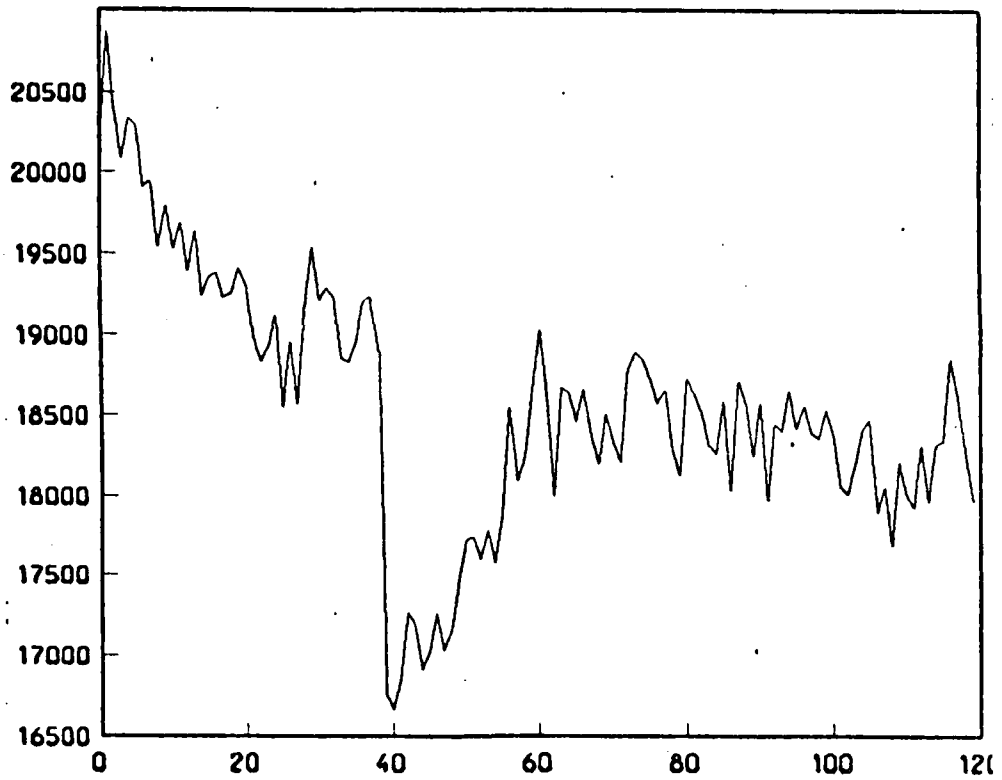
DAY 5, DIFFERENTIAL TEST, ROW 2, COLUMN 4



DAY 5, DIFFERENTIAL TEST, ROW 3, COLUMN 1

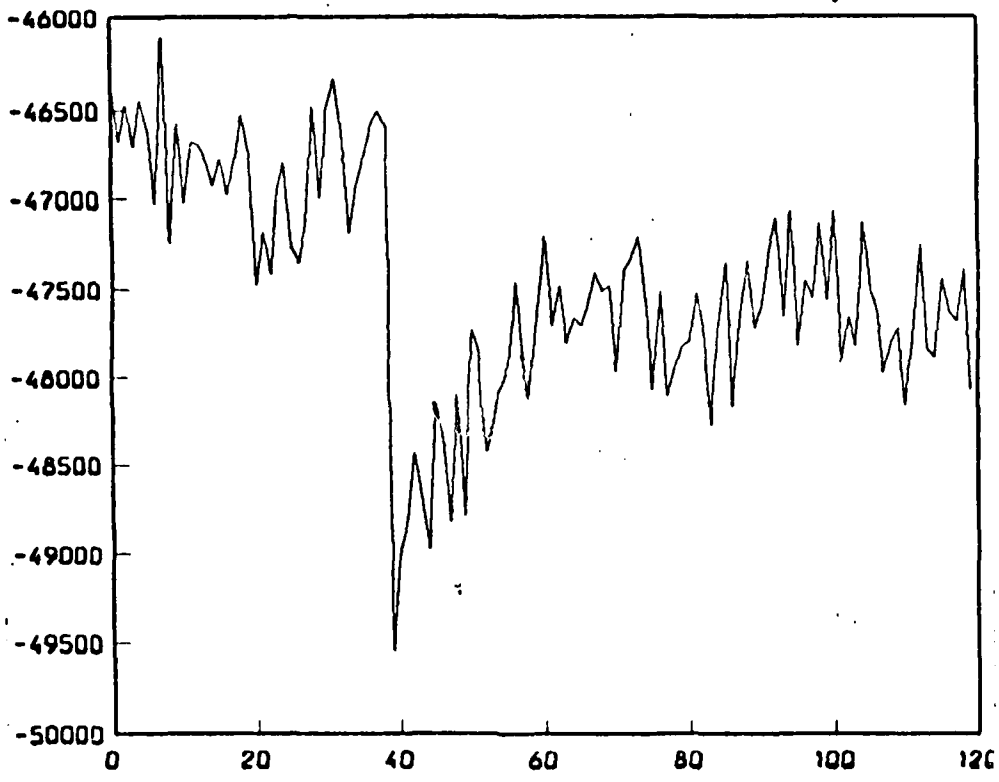


DAY 5, DIFFERENTIAL TEST, ROW 3, COLUMN 2

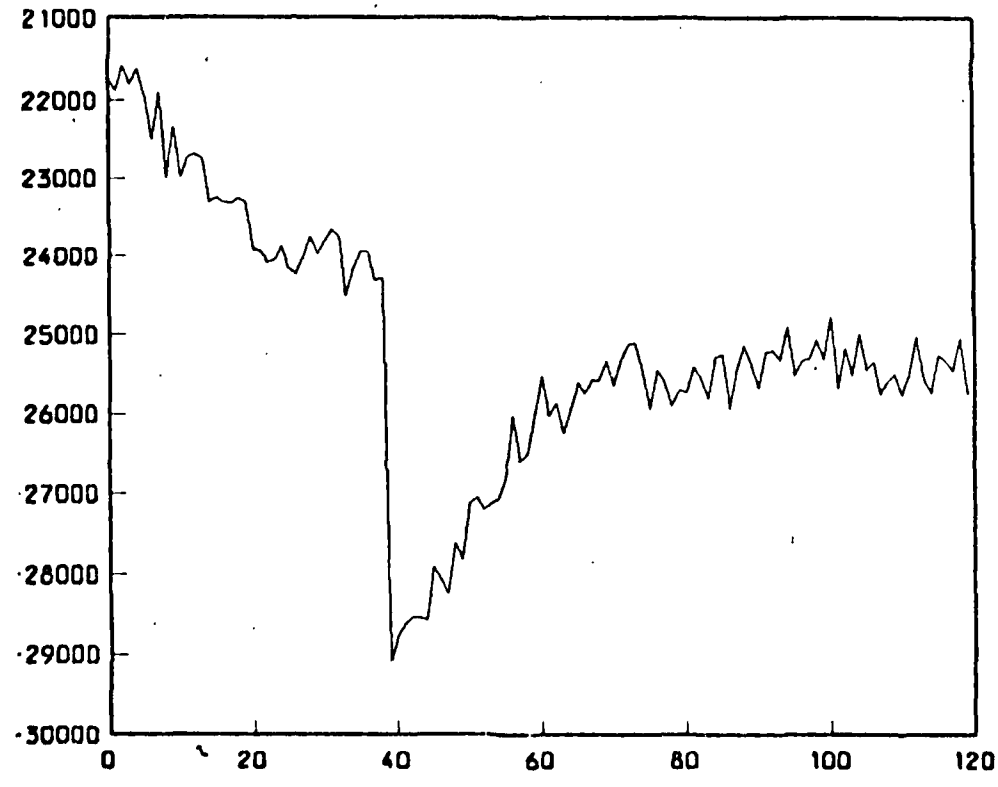


DAY 5, DIFFERENTIAL TEST, ROW 3, COLUMN 3

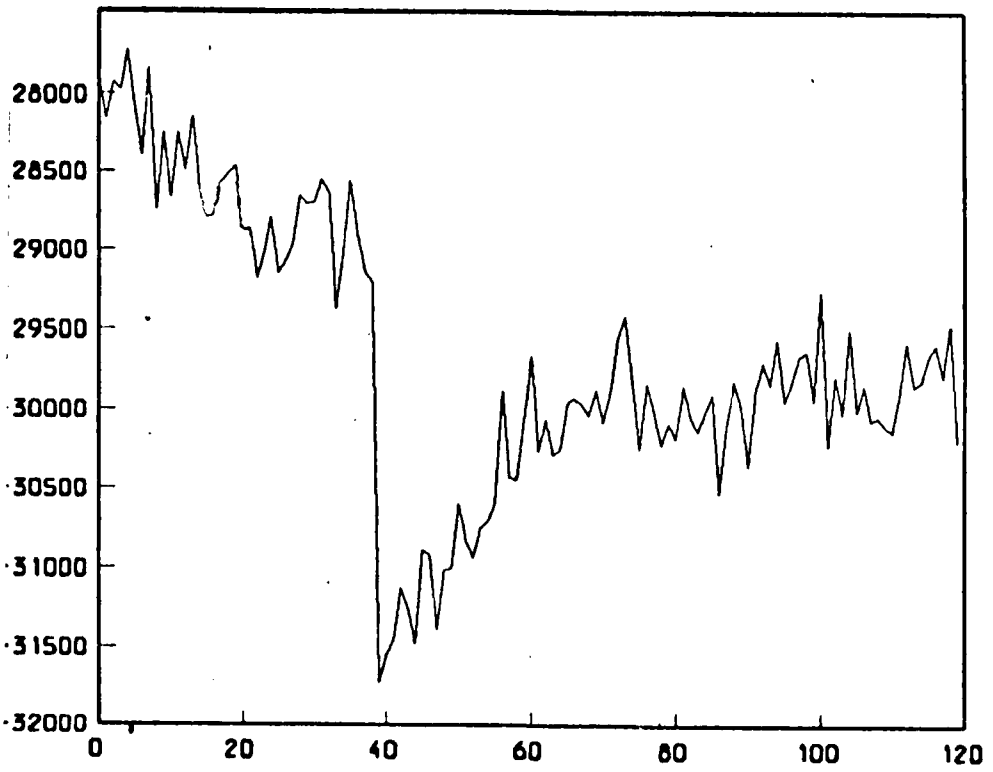
DAY 5, DIFFERENTIAL TEST, ROW 3, COLUMN 4



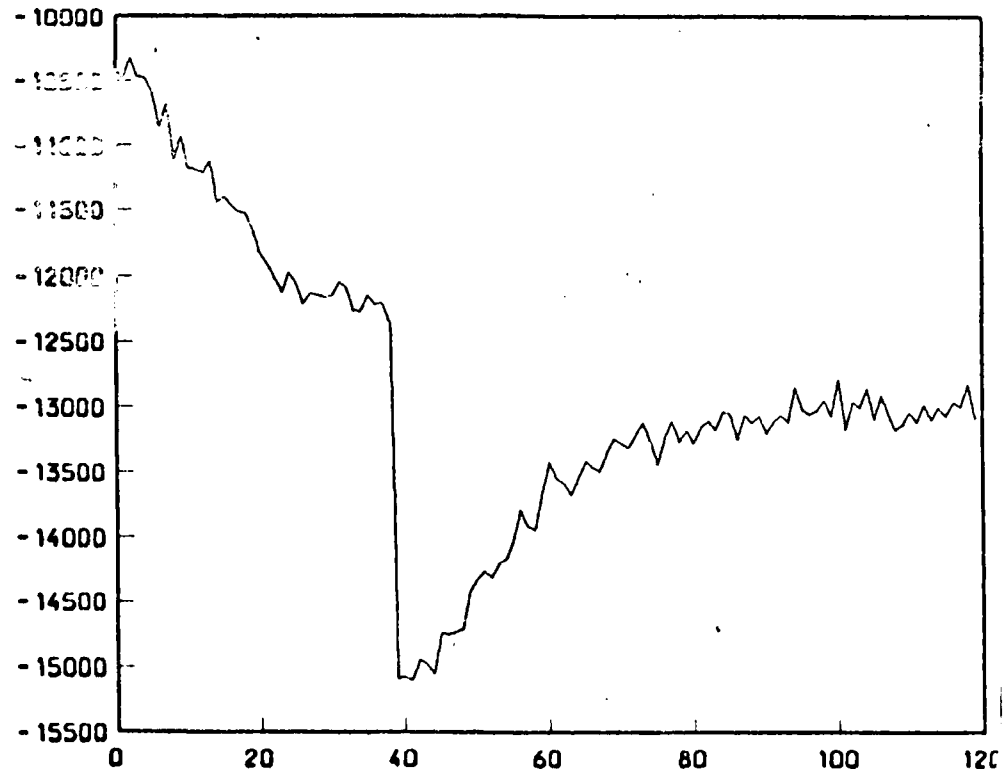
DAY 5, DIFFERENTIAL TEST, ROW 4, COLUMN 1



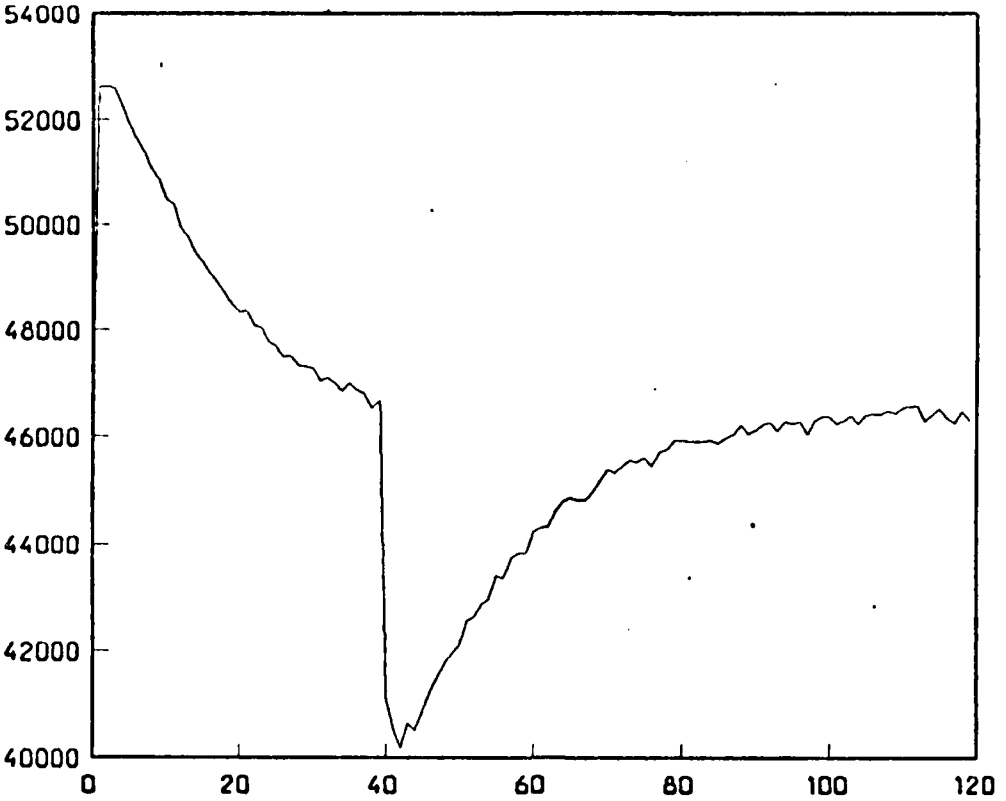
DAY 5, DIFFERENTIAL TEST, ROW 4, COLUMN 2



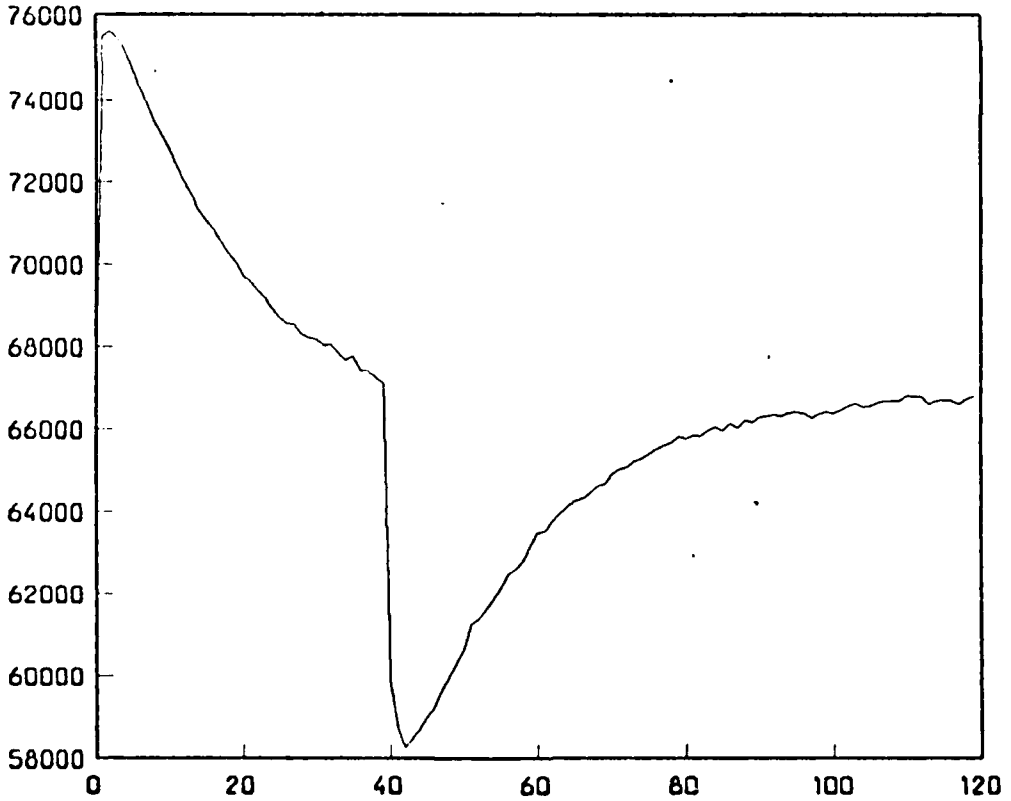
DAY 5, DIFFERENTIAL TEST, ROW 4, COLUMN 3



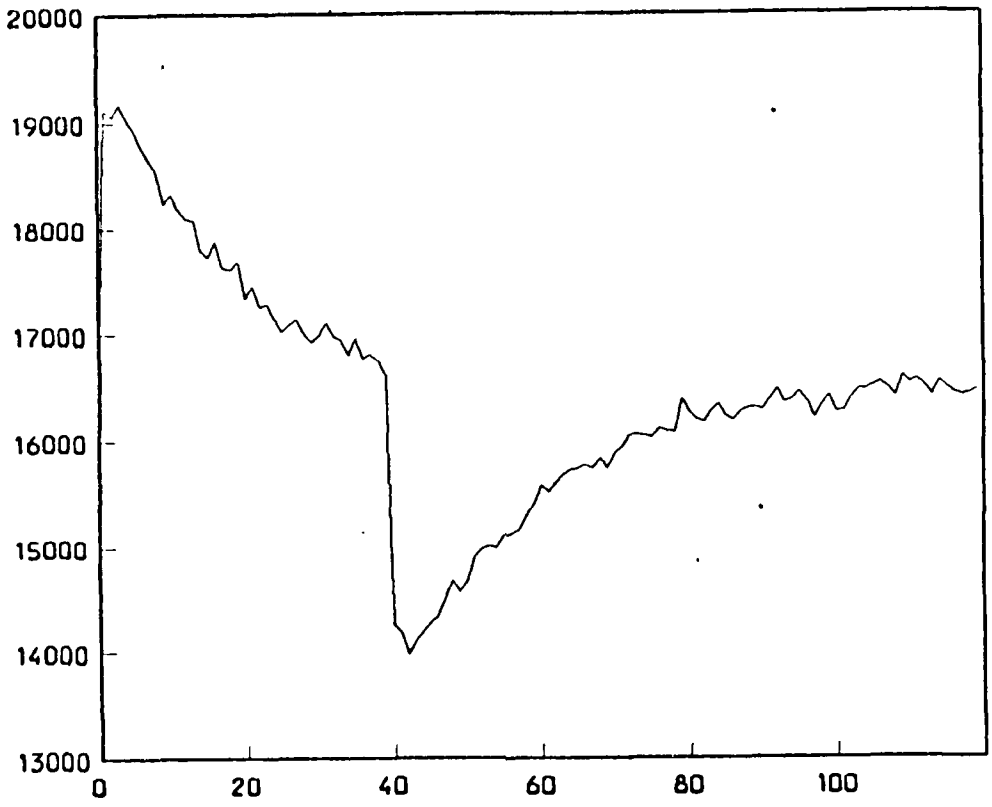
DAY 5, DIFFERENTIAL TEST, ROW 4, COLUMN 4



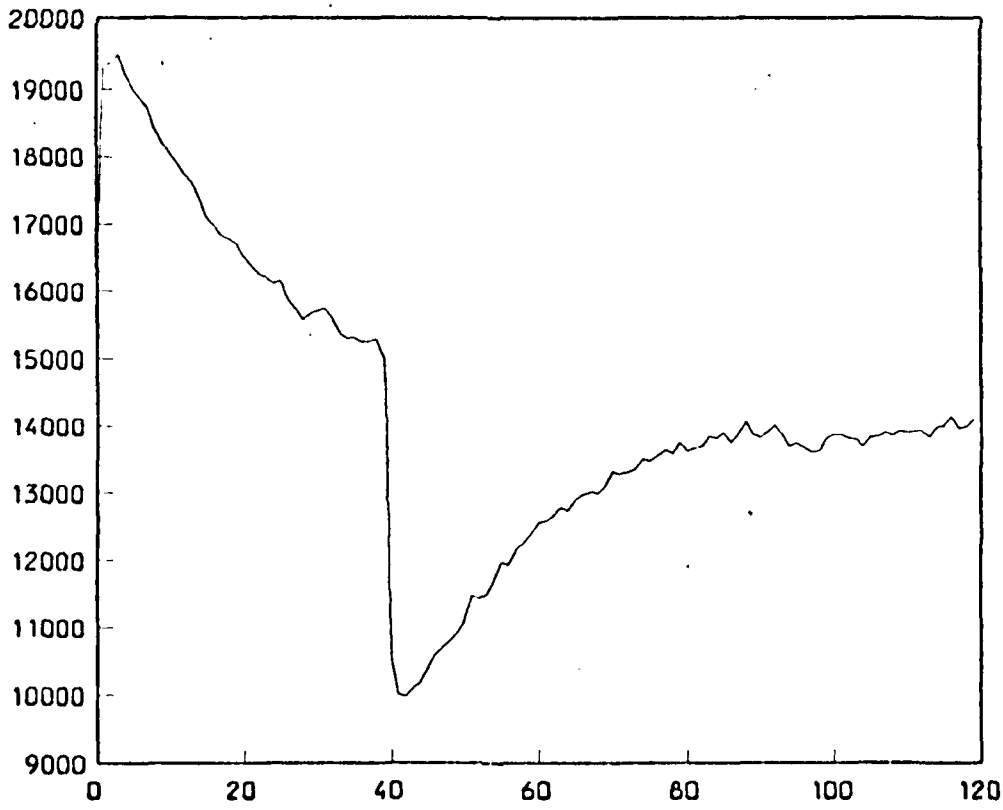
DAY 6, DIFFERENTIAL TEST, ROW 1, COLUMN 1



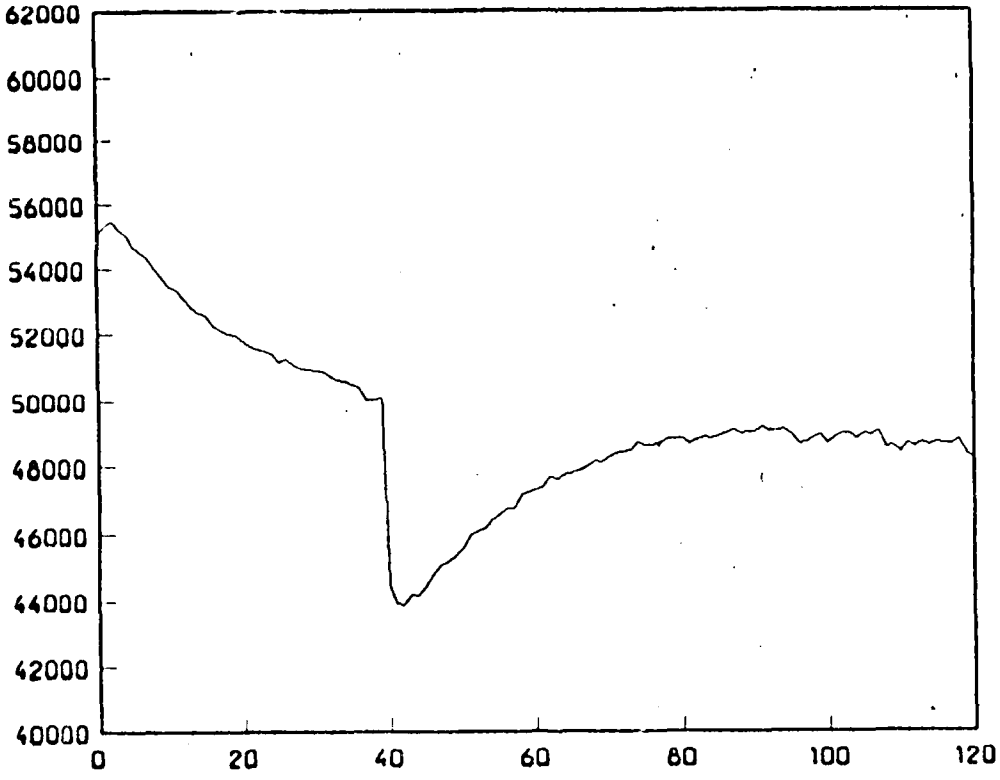
DAY 6, DIFFERENTIAL TEST, ROW 1, COLUMN 2



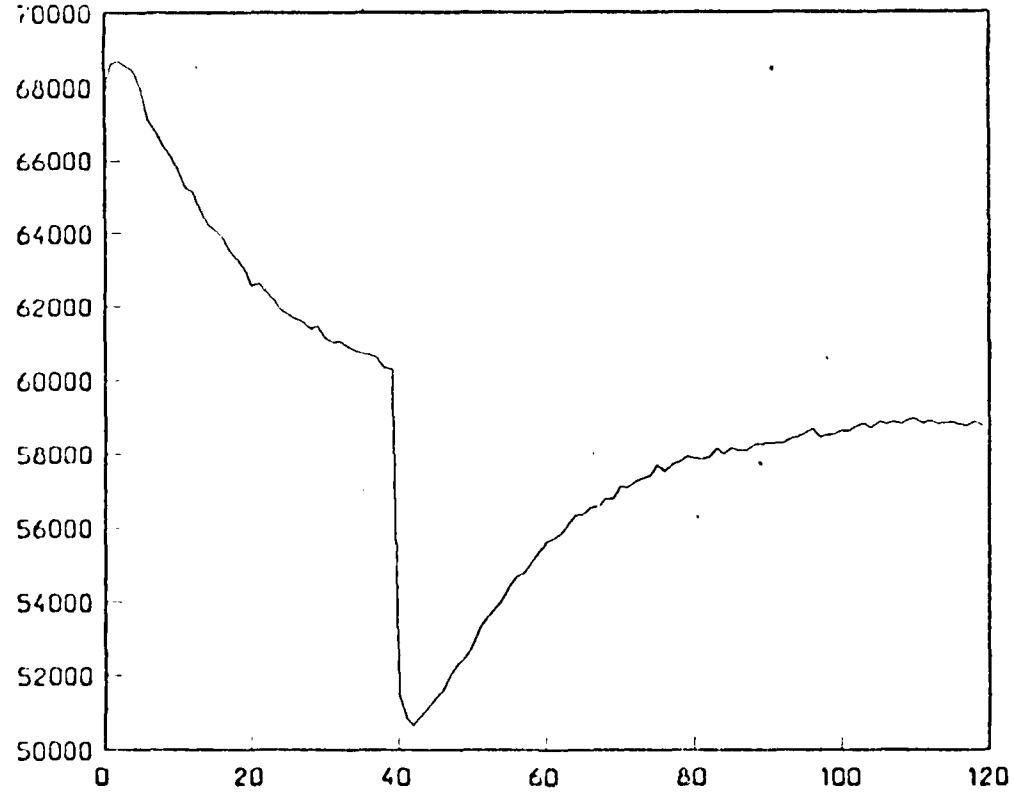
DAY 6, DIFFERENTIAL TEST, ROW 1, COLUMN 3



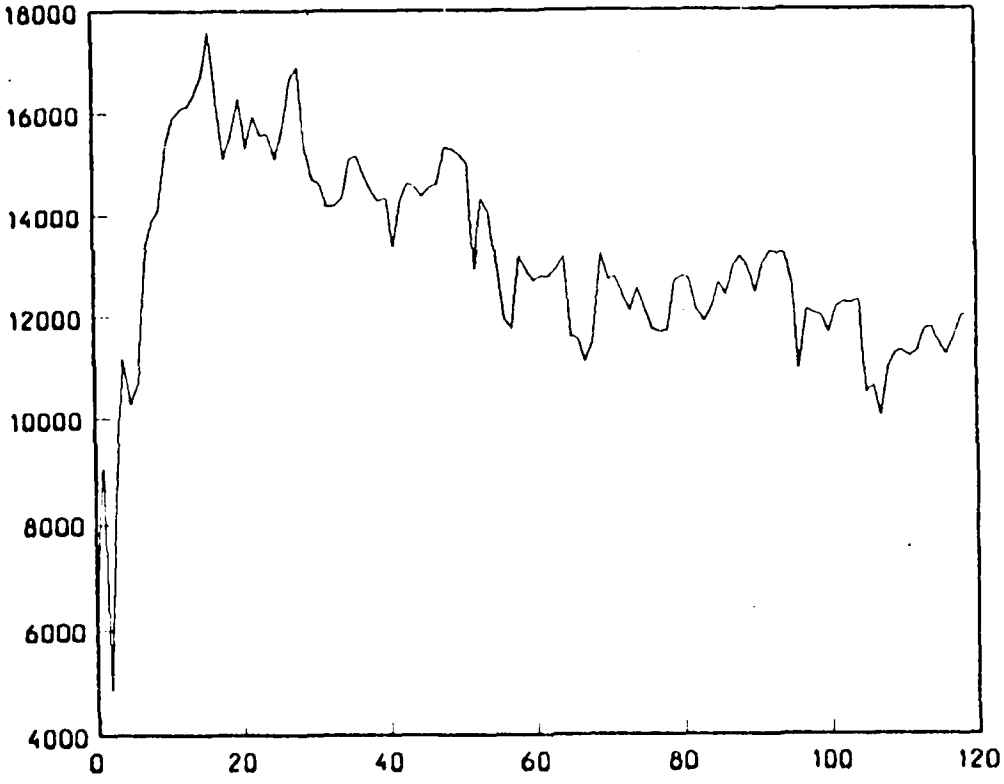
DAY 6, DIFFERENTIAL TEST, ROW 1, COLUMN 4



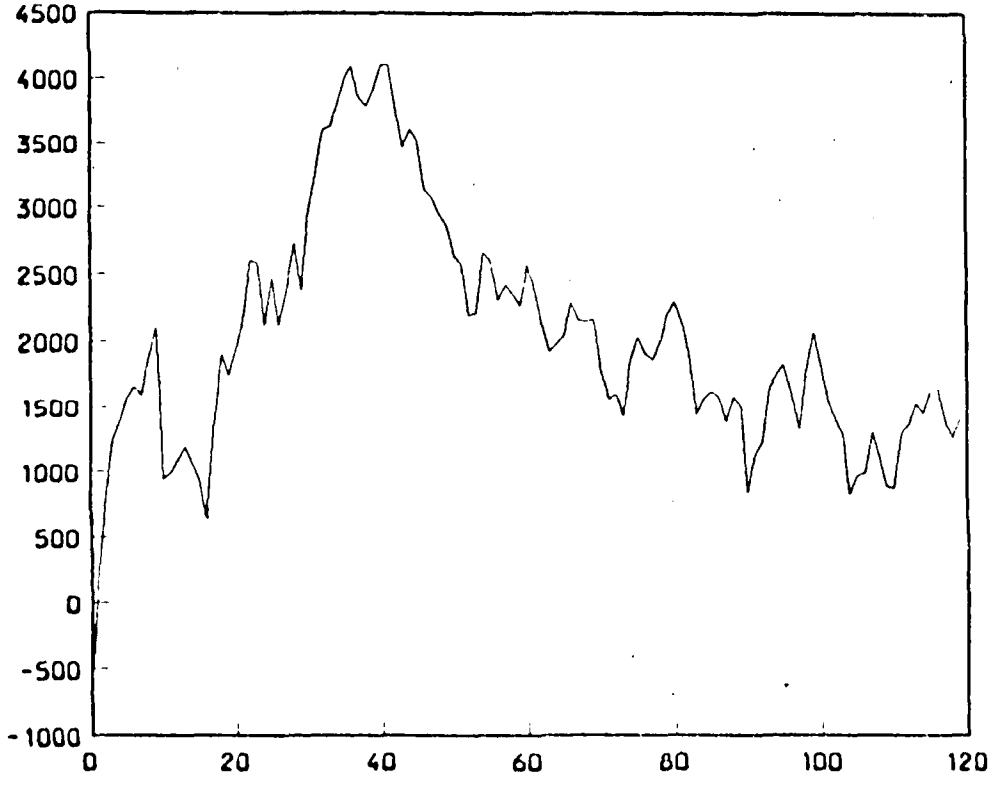
DAY 6, DIFFERENTIAL TEST, ROW 2, COLUMN 1



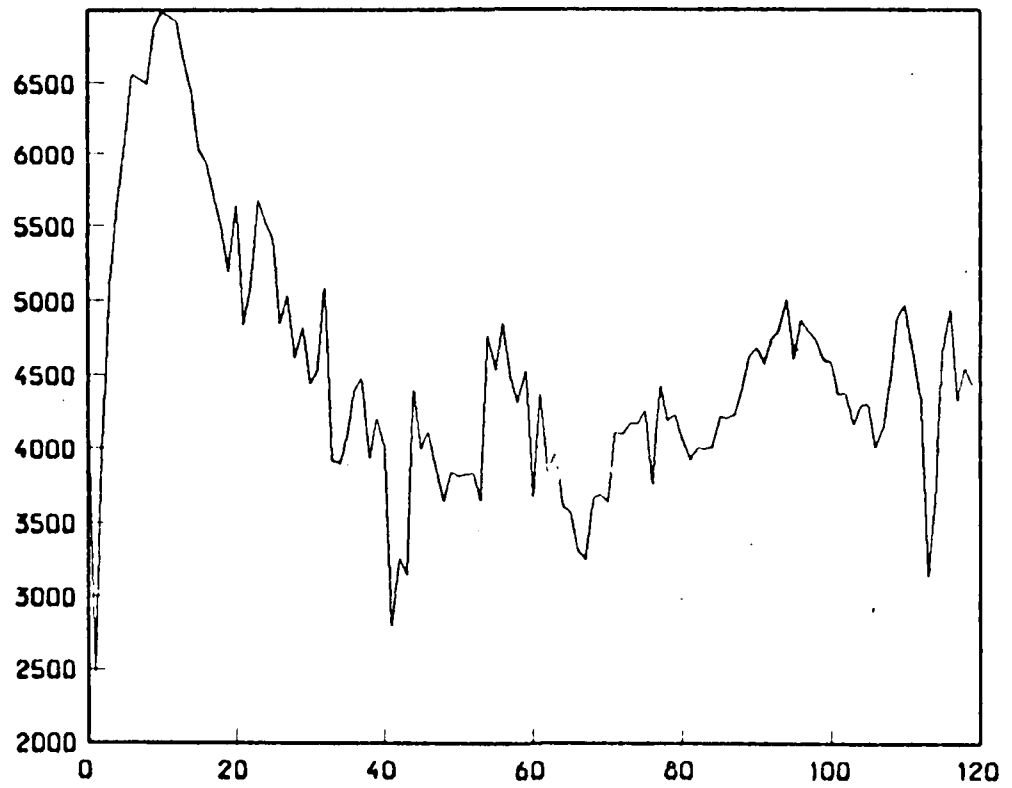
DAY 6, DIFFERENTIAL TEST, ROW 2, COLUMN 2



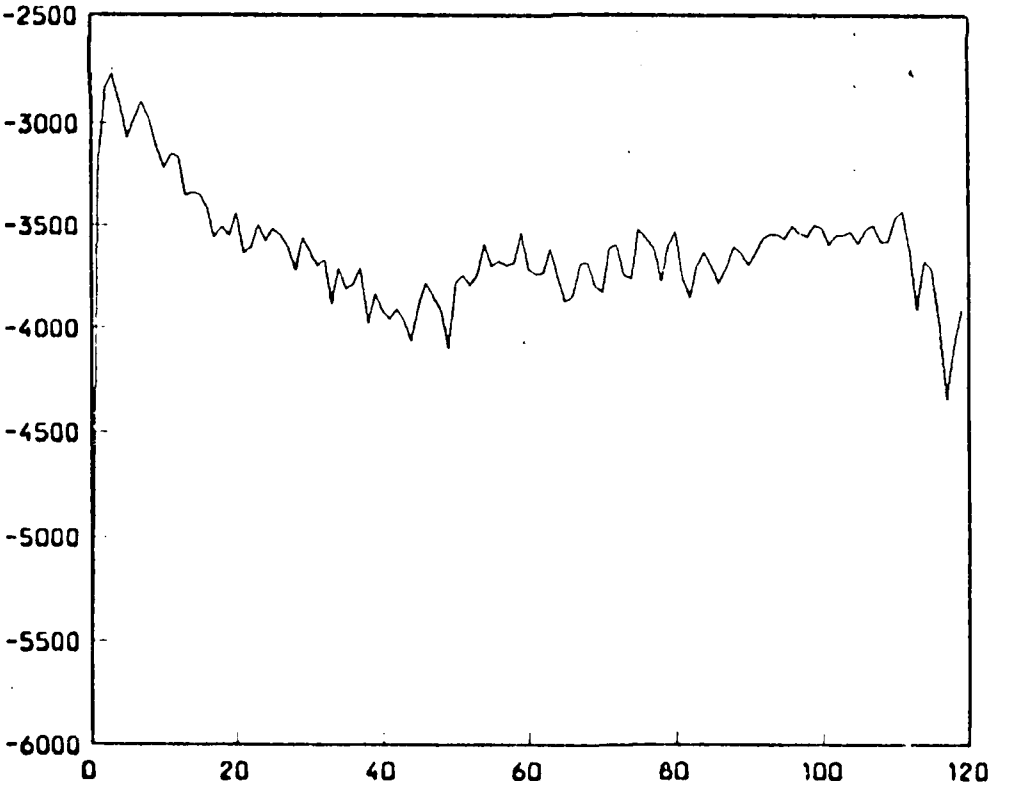
DAY 7, ABSOLUTE TEST, ROW 4, COLUMN 1



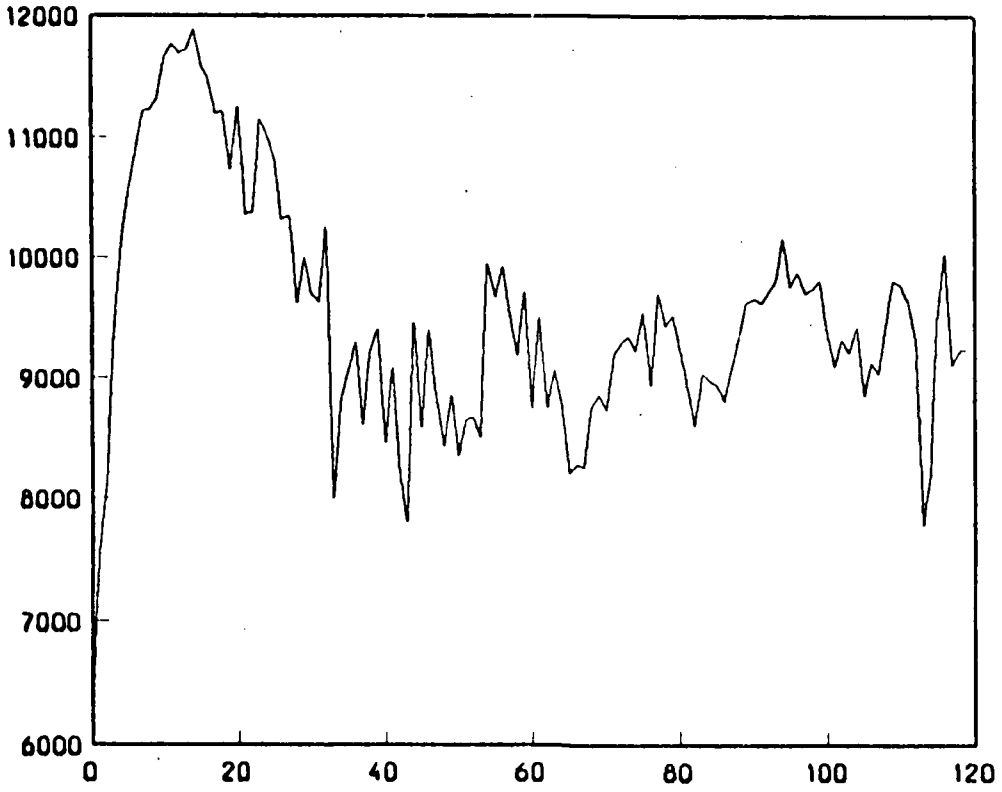
DAY 7, ABSOLUTE TEST, ROW 4, COLUMN 2



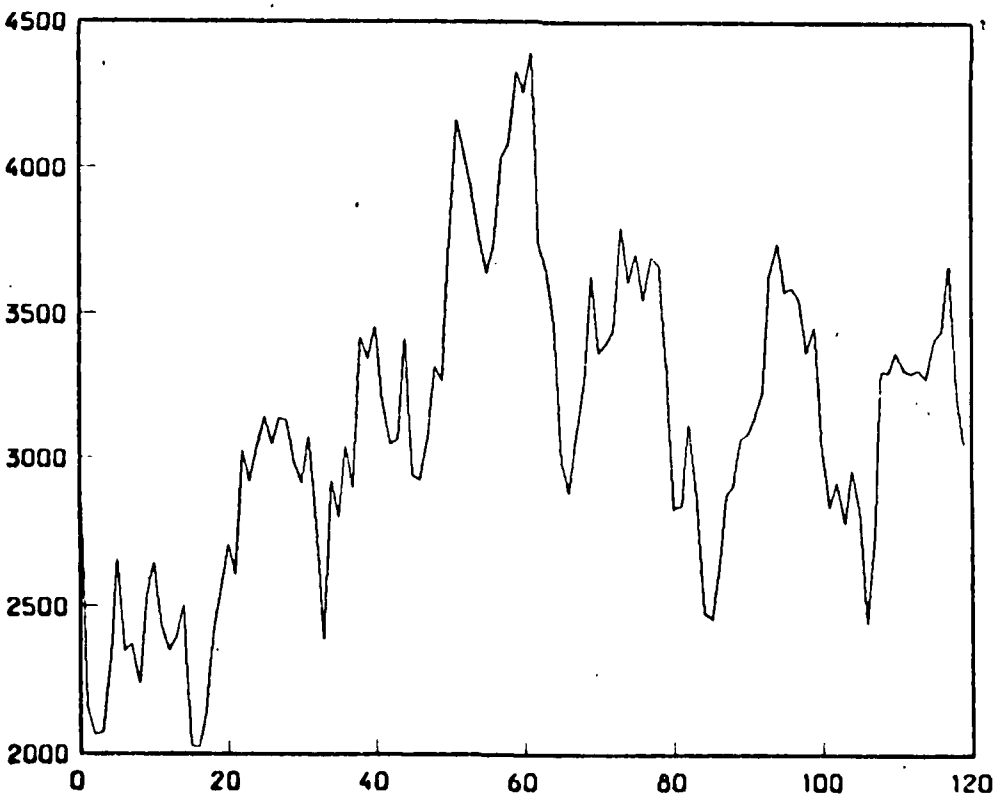
DAY 7, DIFFERENTIAL TEST, ROW 4, COLUMN 3, WITH BUCKET



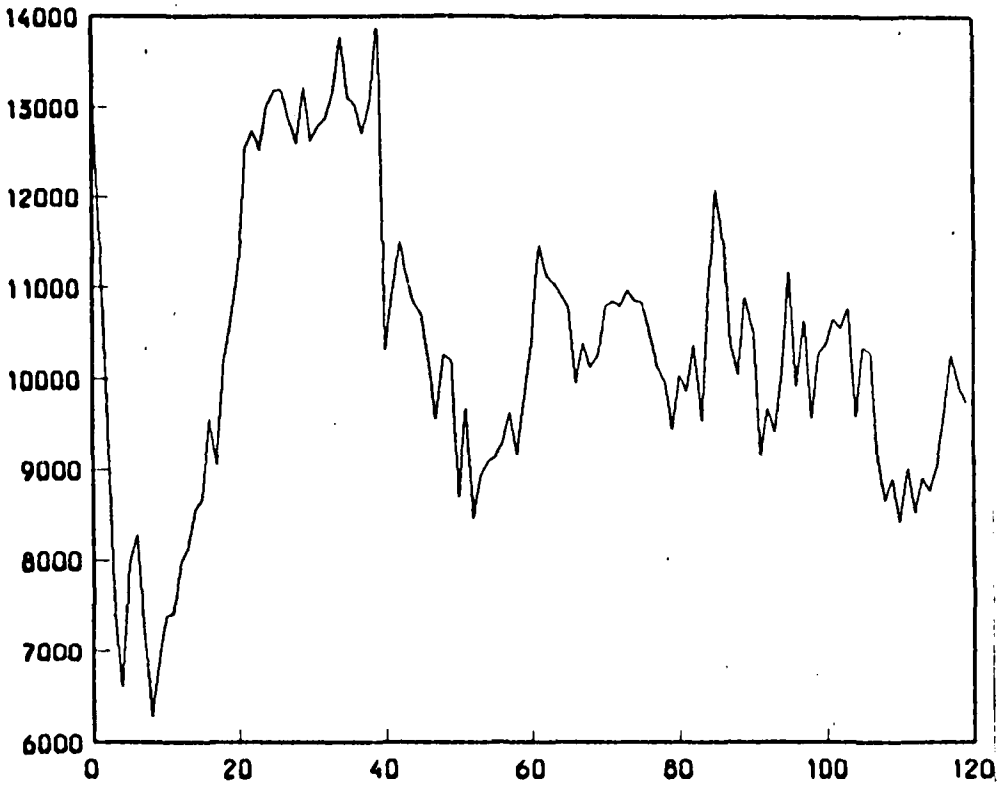
DAY 7, DIFFERENTIAL TEST, ROW 4, COLUMN 4, WITH BUCKET



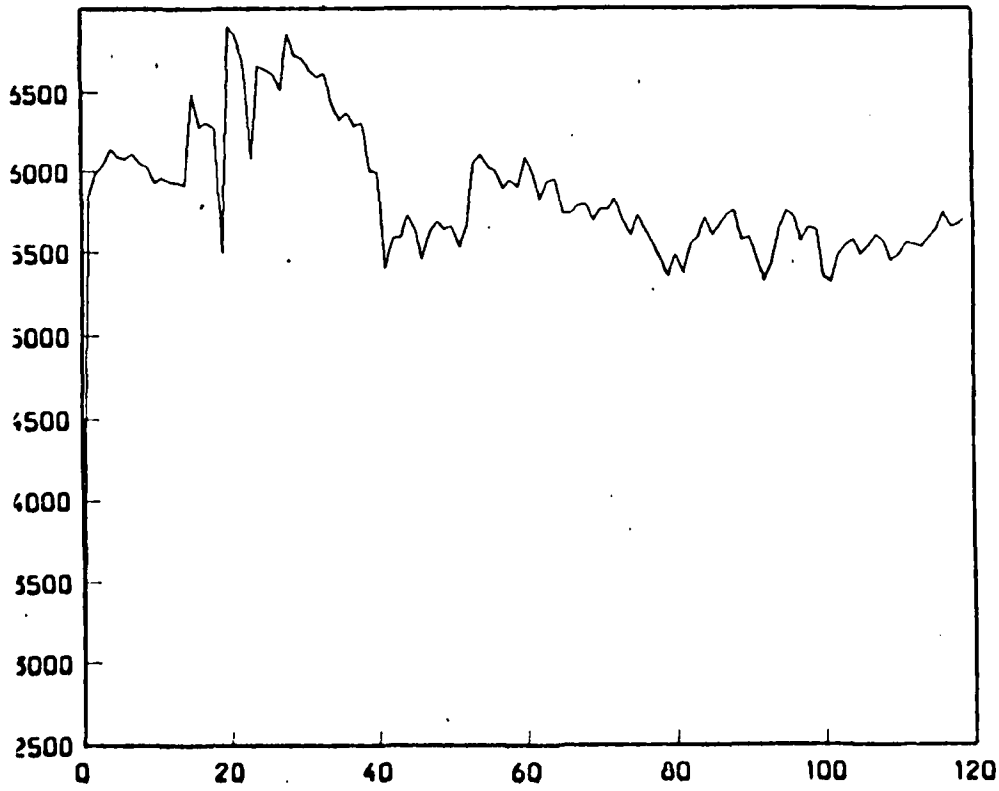
DAY 7, DIFFERENTIAL TEST, ROW 4, COLUMN 1, WITH BUCKET



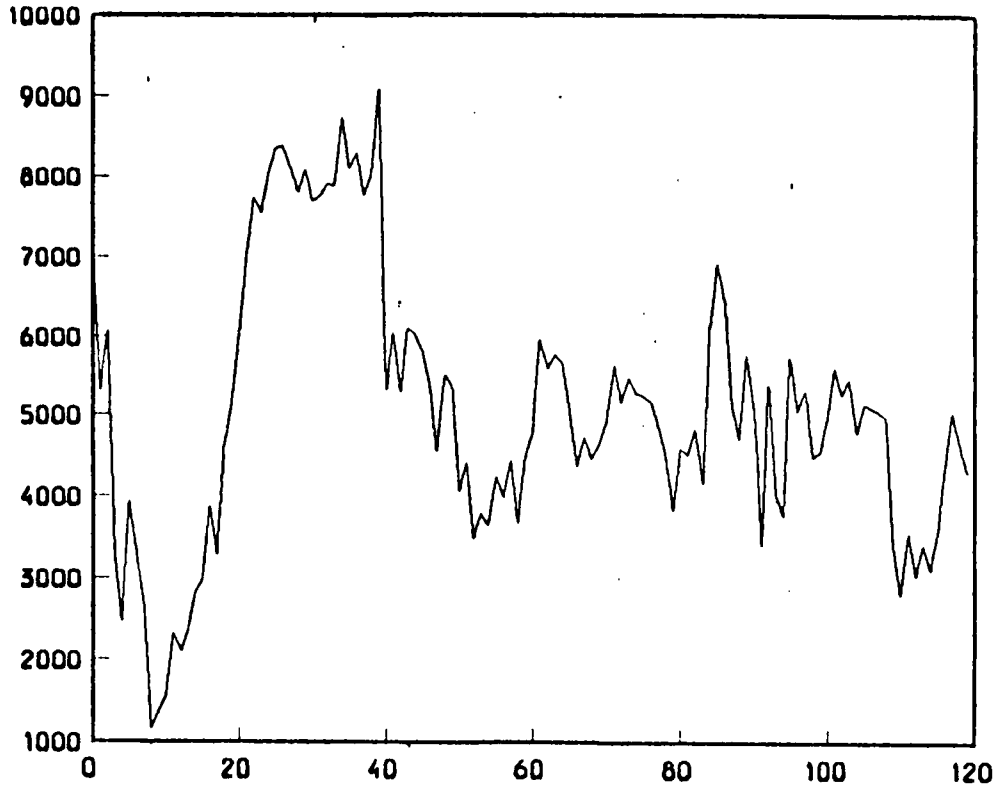
DAY 7, DIFFERENTIAL TEST, ROW 4, COLUMN 2, WITH BUCKET



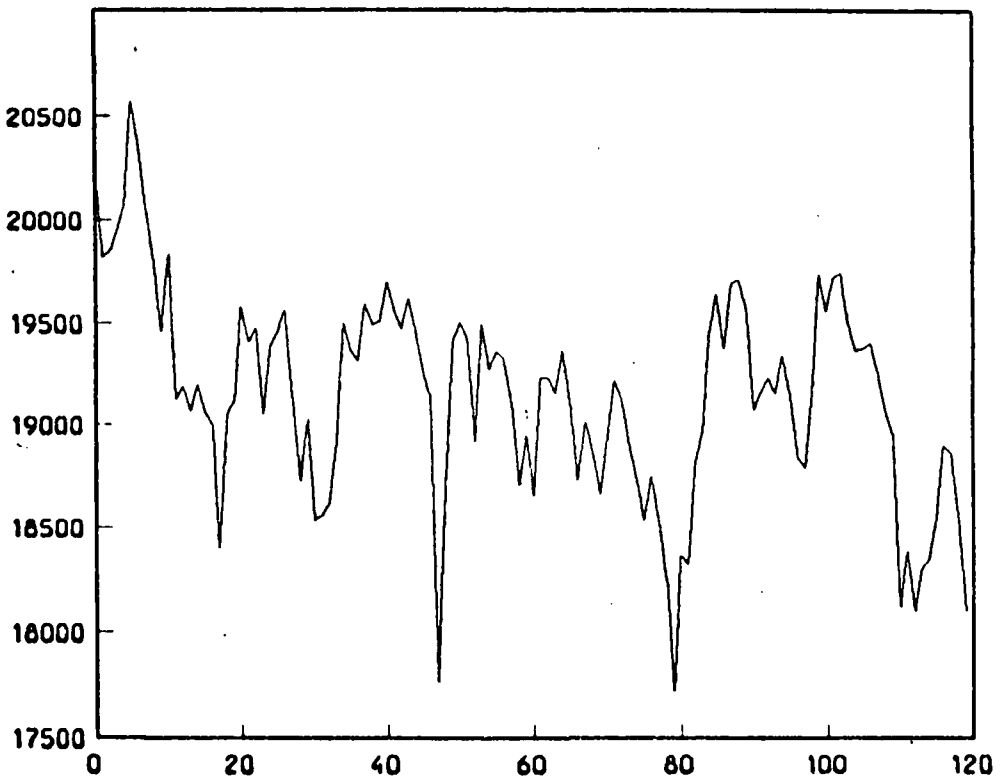
DAY 7, DIFFERENTIAL TEST, ROW 4, COLUMN 3



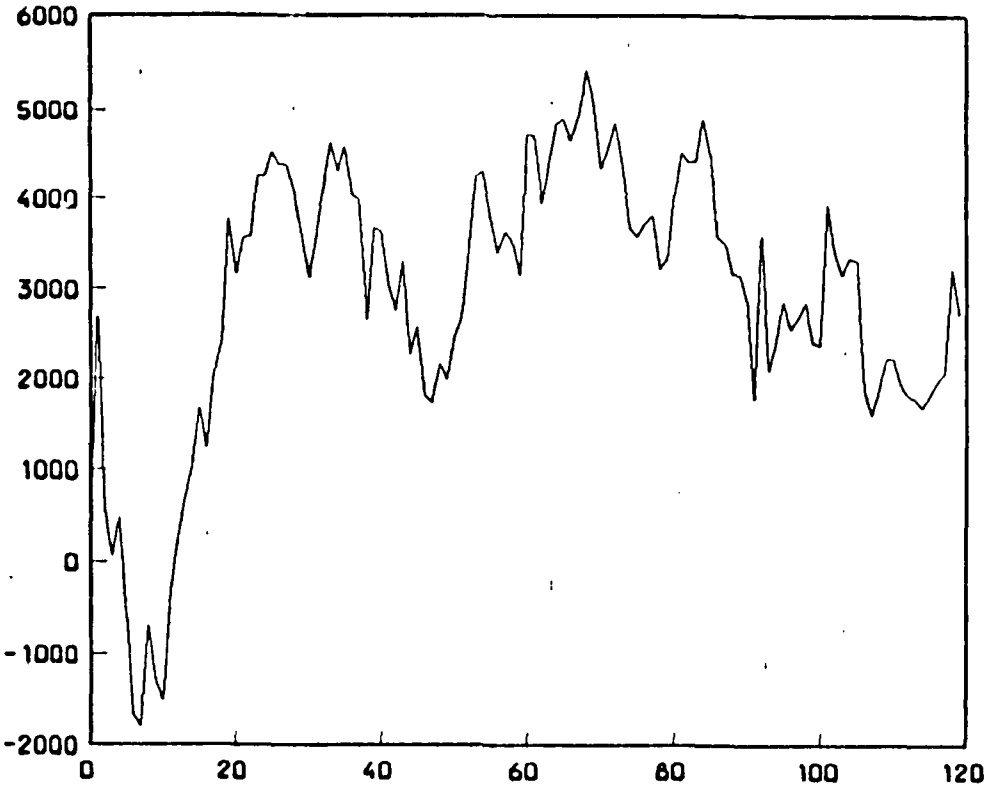
DAY 7, DIFFERENTIAL TEST, ROW 4, COLUMN 4



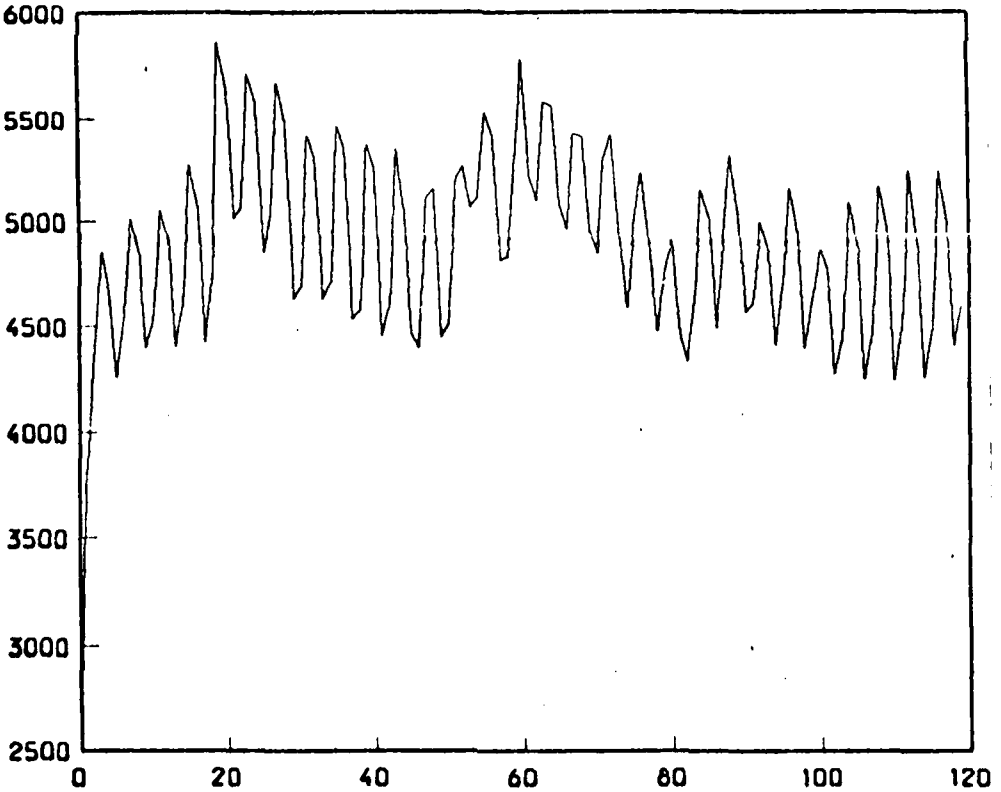
DAY 7, DIFFERENTIAL TEST, ROW 4, COLUMN 1



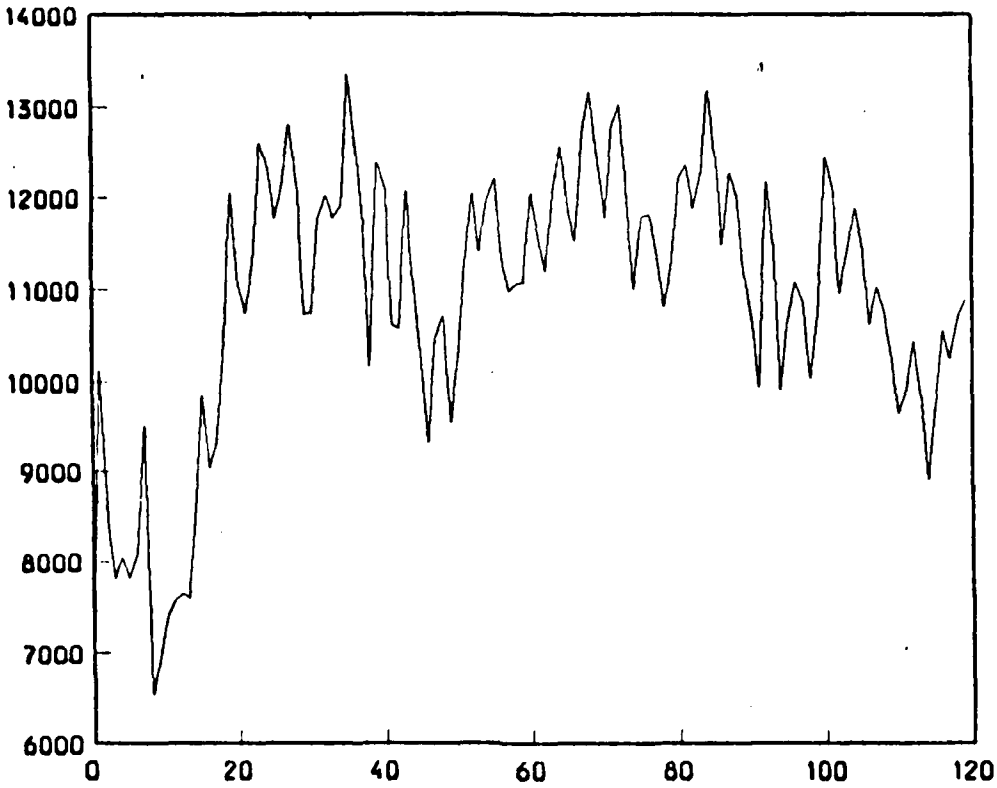
DAY 7, DIFFERENTIAL TEST, ROW 4, COLUMN 2



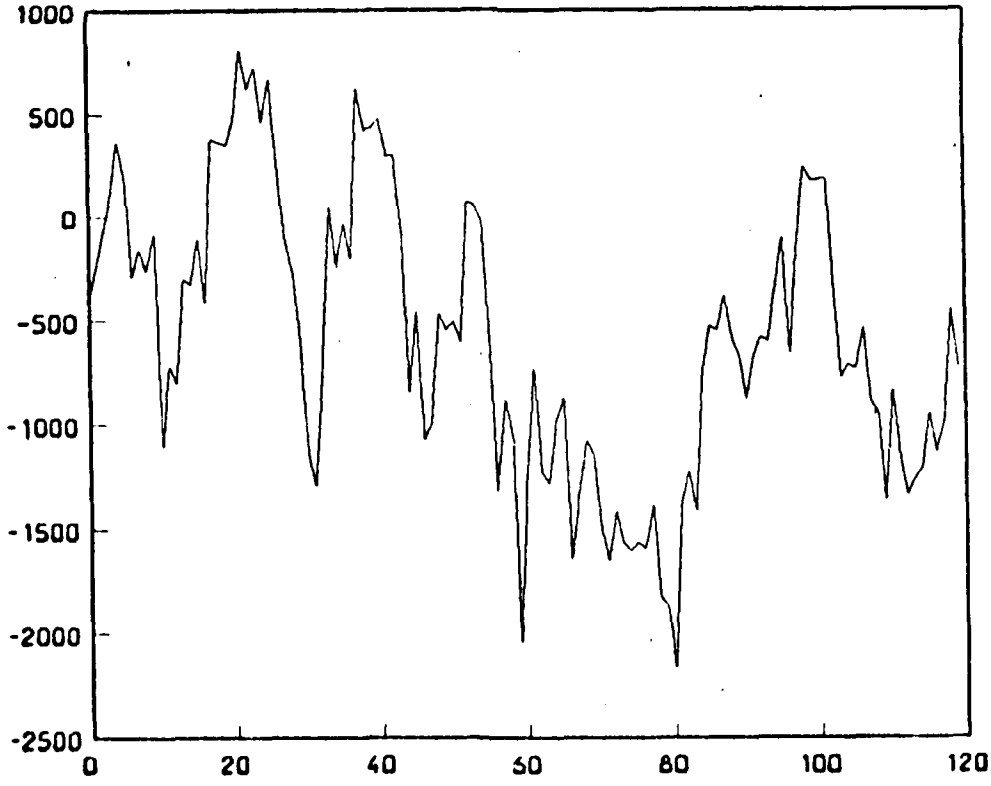
DAY 7, DIFFERENTIAL TEST, ROW 3, COLUMN 3



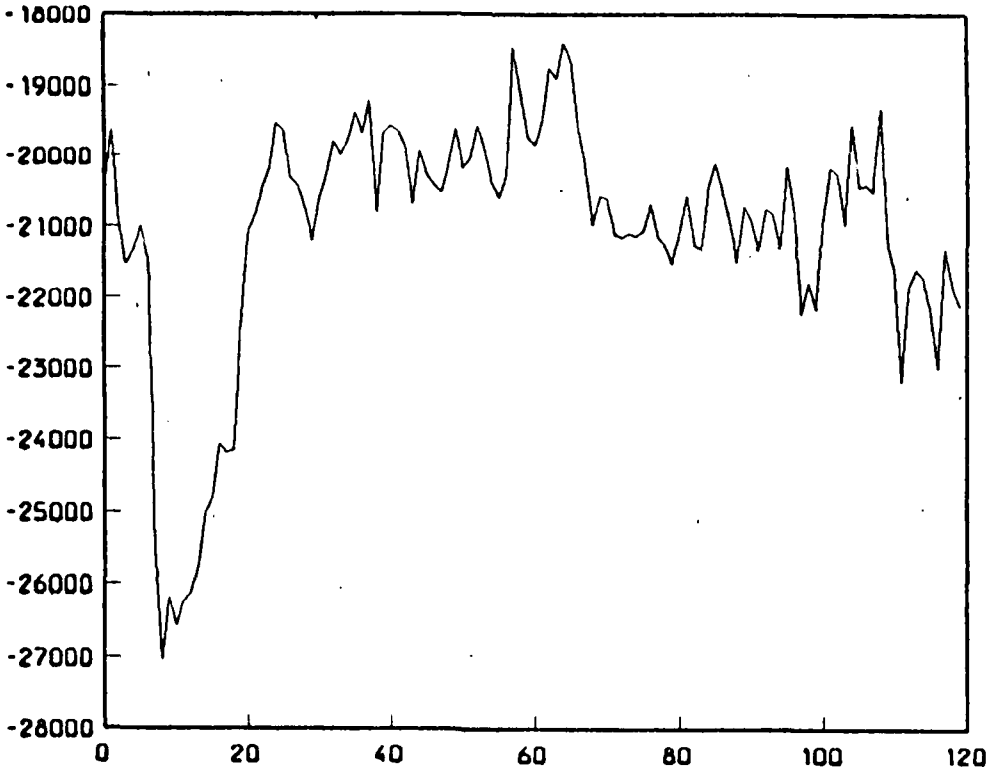
DAY 7, DIFFERENTIAL TEST, ROW 3, COLUMN 4



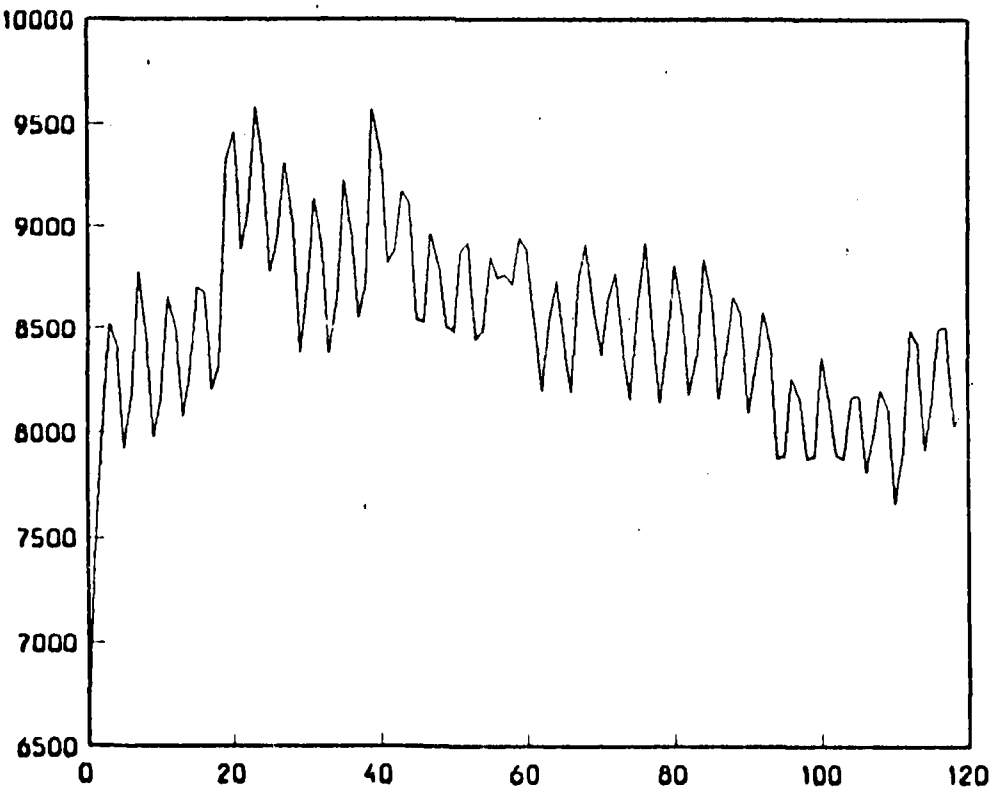
DAY 7, DIFFERENTIAL TEST, ROW 3, COLUMN 1



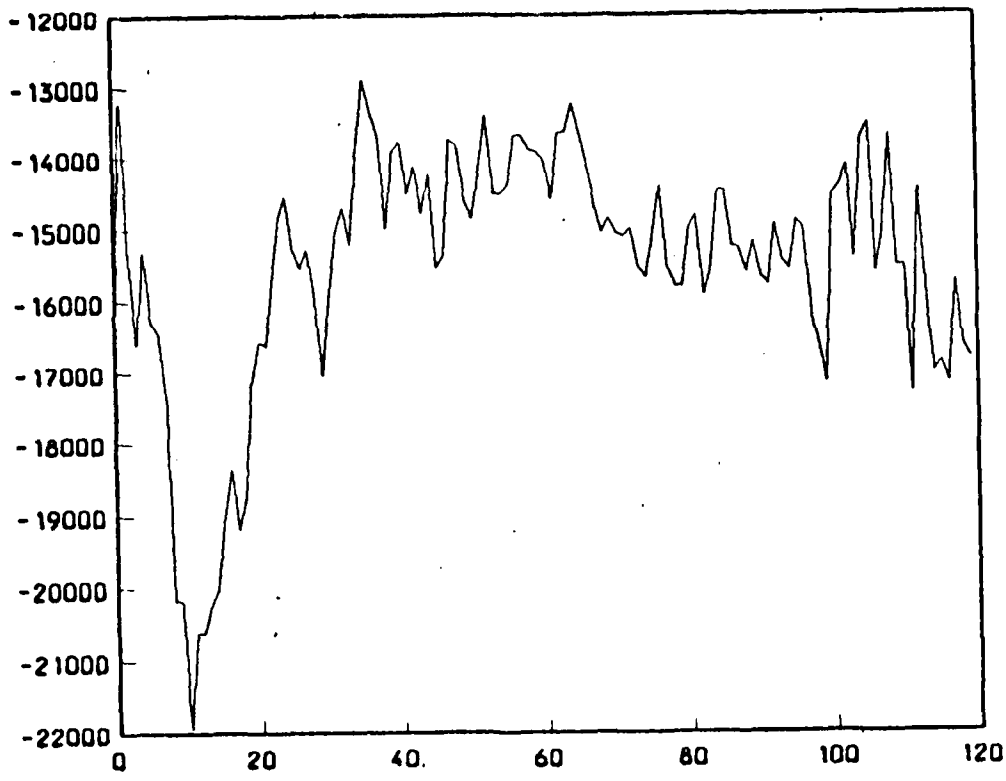
DAY 7, DIFFERENTIAL TEST, ROW 3, COLUMN 2



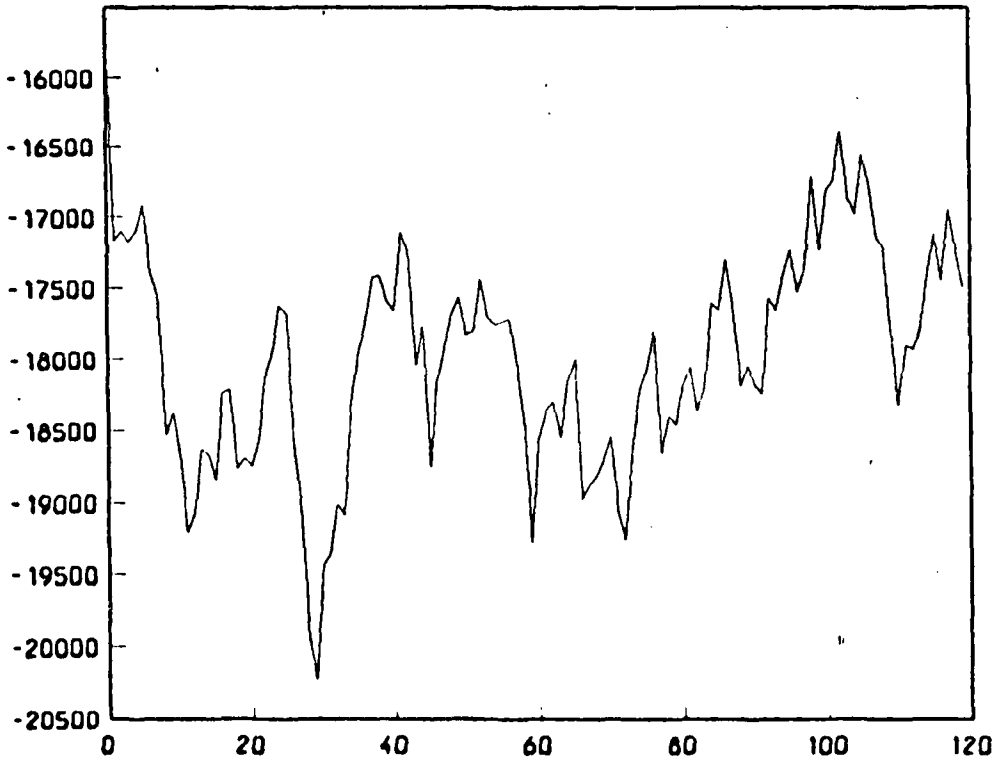
DAY 7, DIFFERENTIAL TEST, ROW 2, COLUMN 3



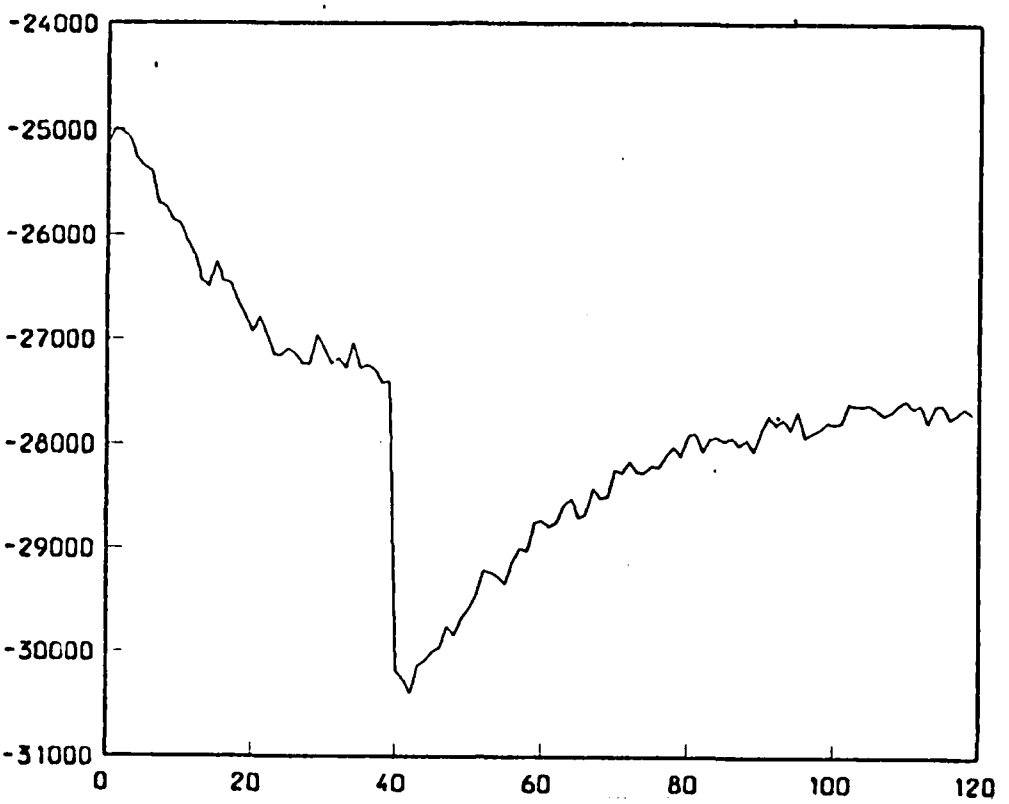
DAY 7, DIFFERENTIAL TEST, ROW 2, COLUMN 4



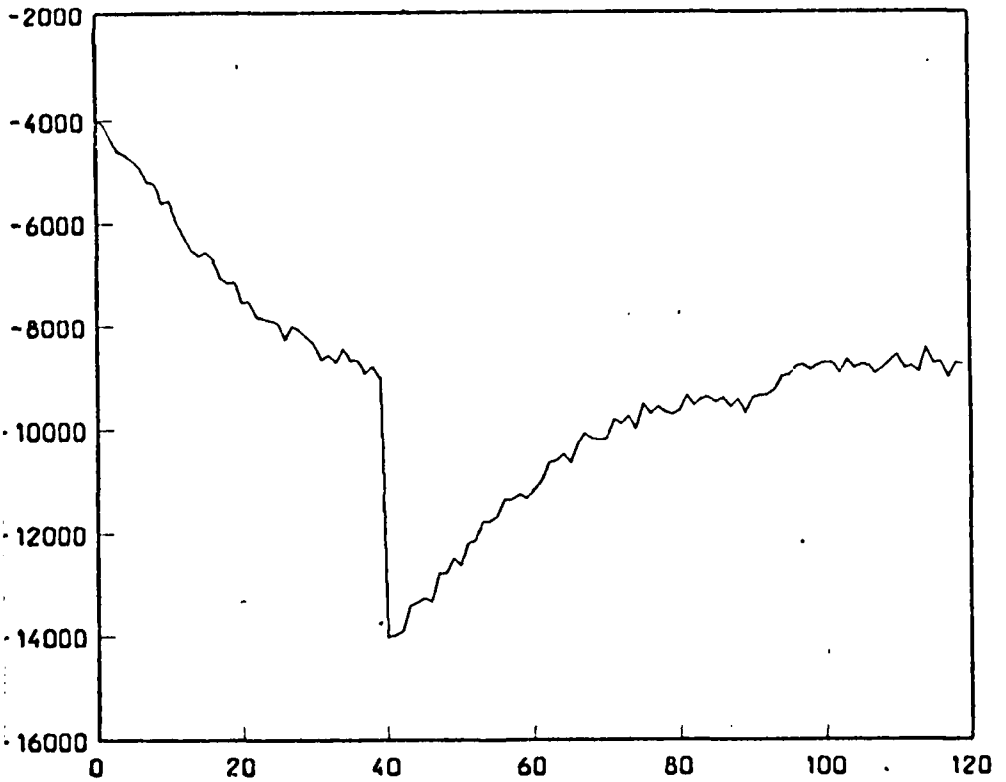
DAY 7, DIFFERENTIAL TEST, ROW 2, COLUMN 1



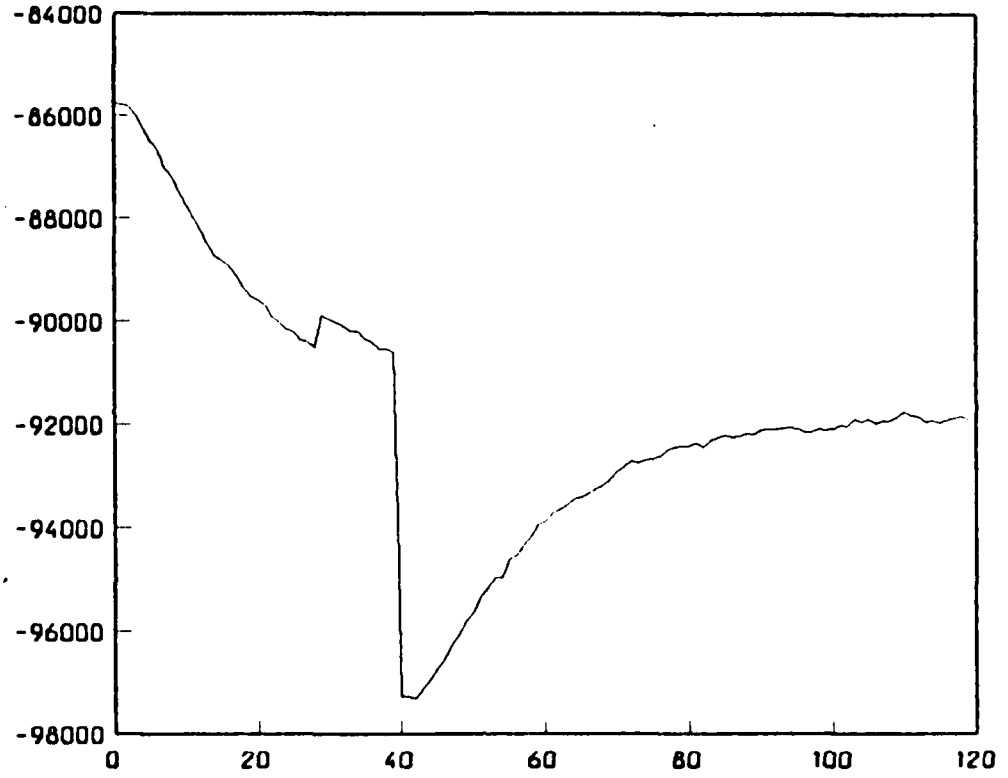
DAY 7, DIFFERENTIAL TEST, ROW 2, COLUMN 2



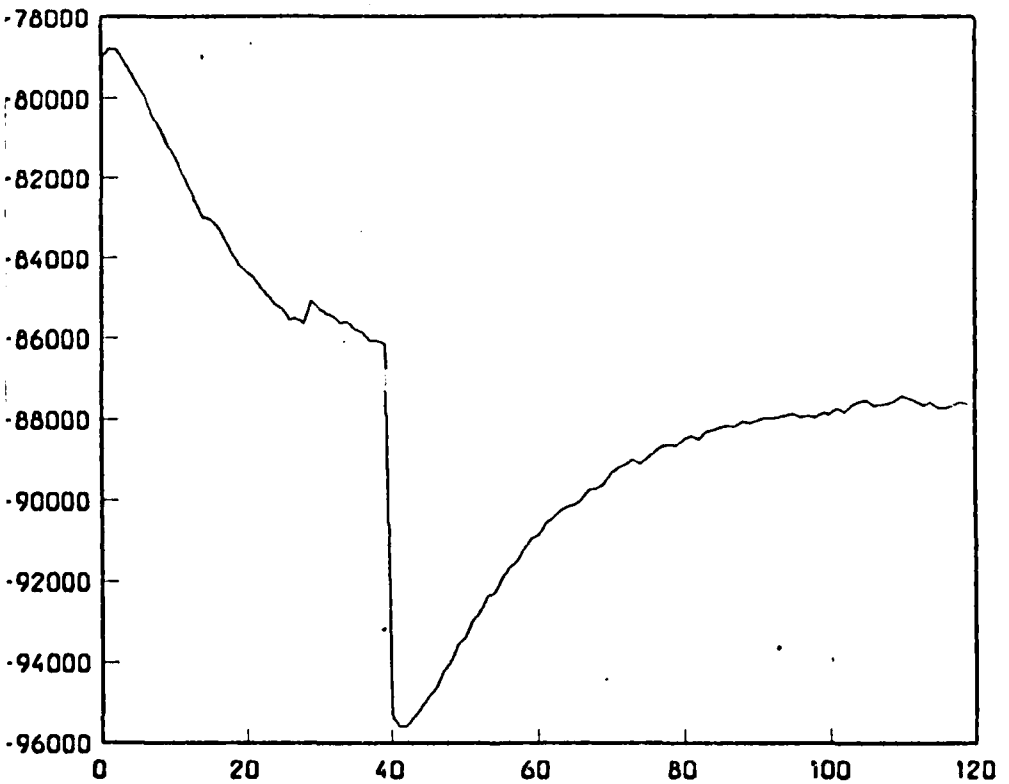
DAY 6, DIFFERENTIAL TEST, ROW 4, COLUMN 3



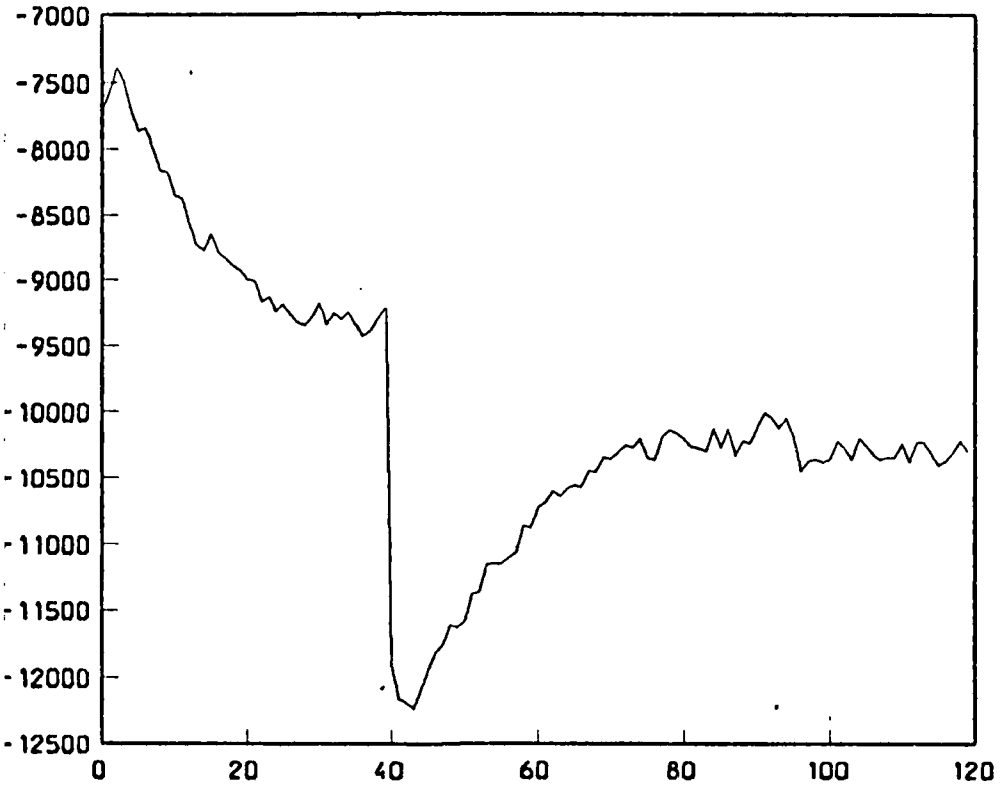
DAY 6, DIFFERENTIAL TEST, ROW 4, COLUMN 4



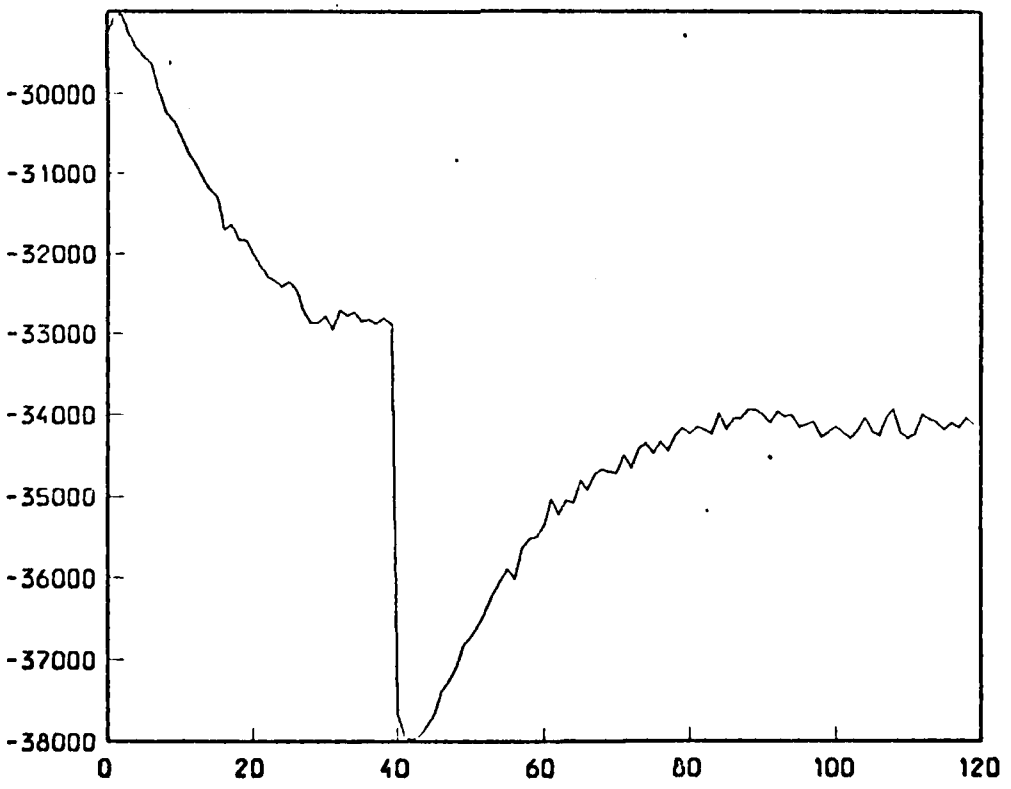
DAY 6, DIFFERENTIAL TEST, ROW 4, COLUMN 1



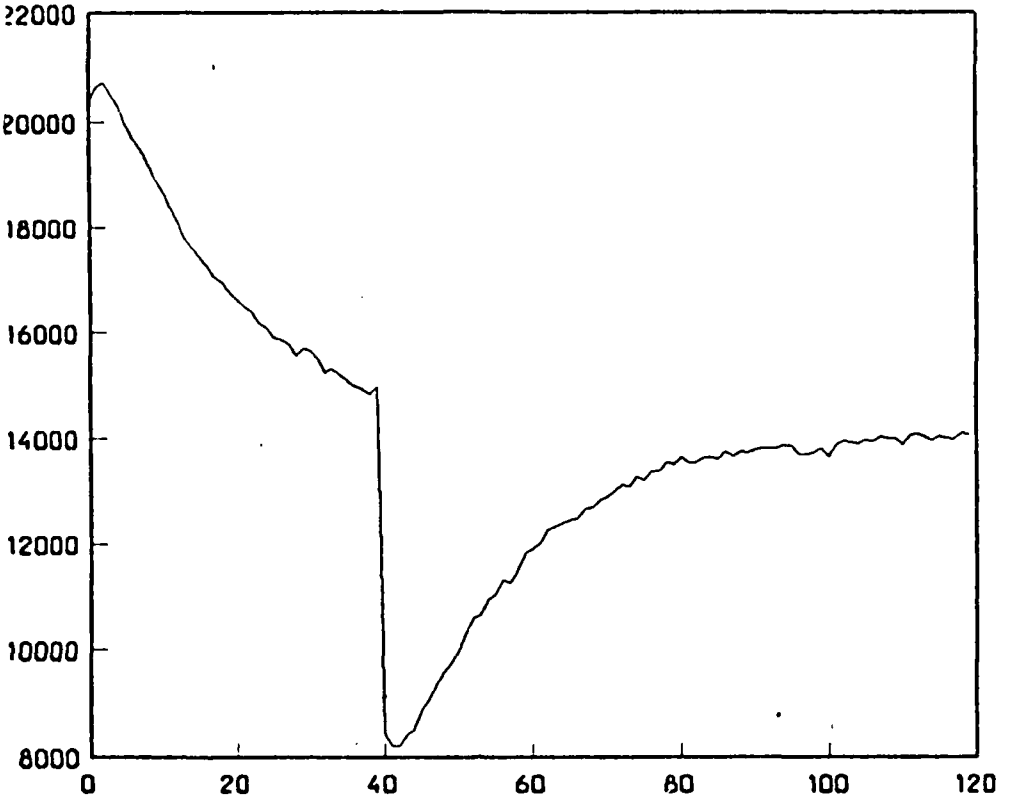
DAY 6, DIFFERENTIAL TEST, ROW 4, COLUMN 2



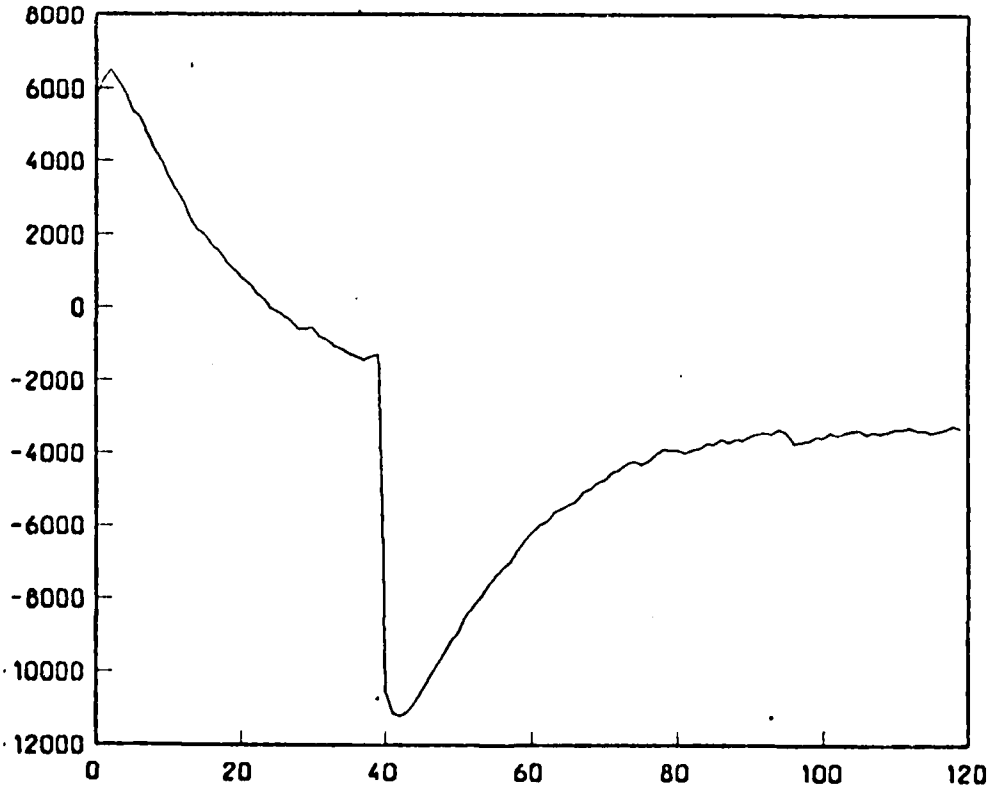
DAY 6, DIFFERENTIAL TEST, ROW 3, COLUMN 3



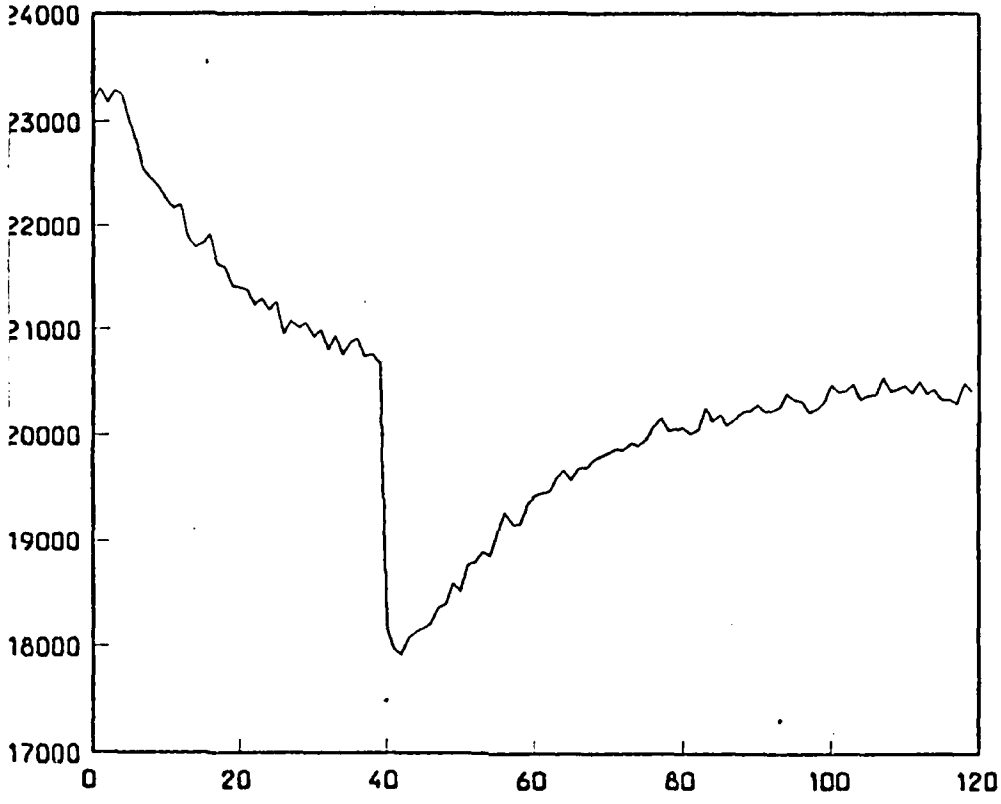
DAY 6, DIFFERENTIAL TEST, ROW 3, COLUMN 4



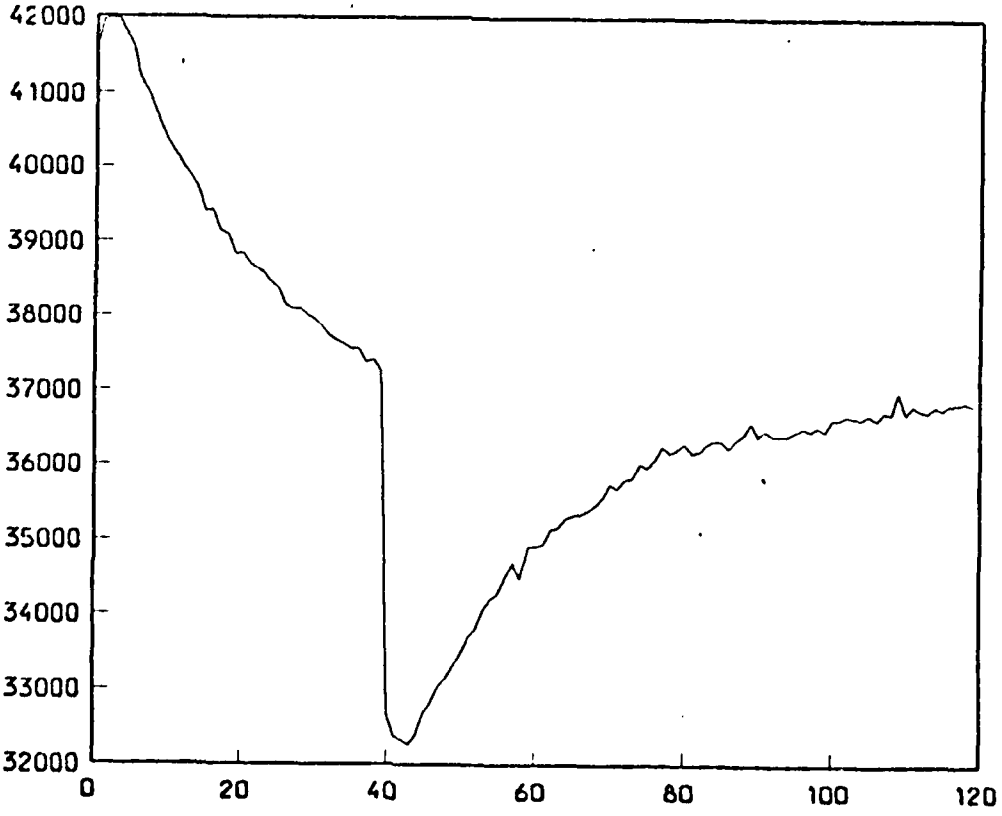
DAY 6, DIFFERENTIAL TEST, ROW 3, COLUMN 1



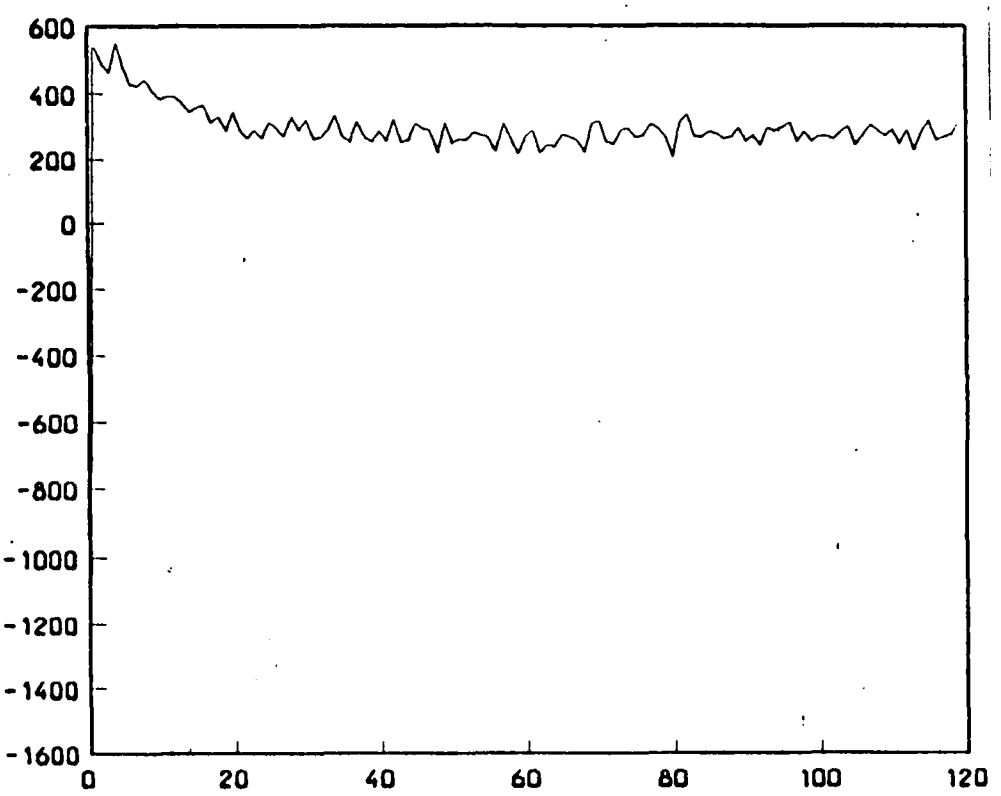
DAY 6, DIFFERENTIAL TEST, ROW 3, COLUMN 2



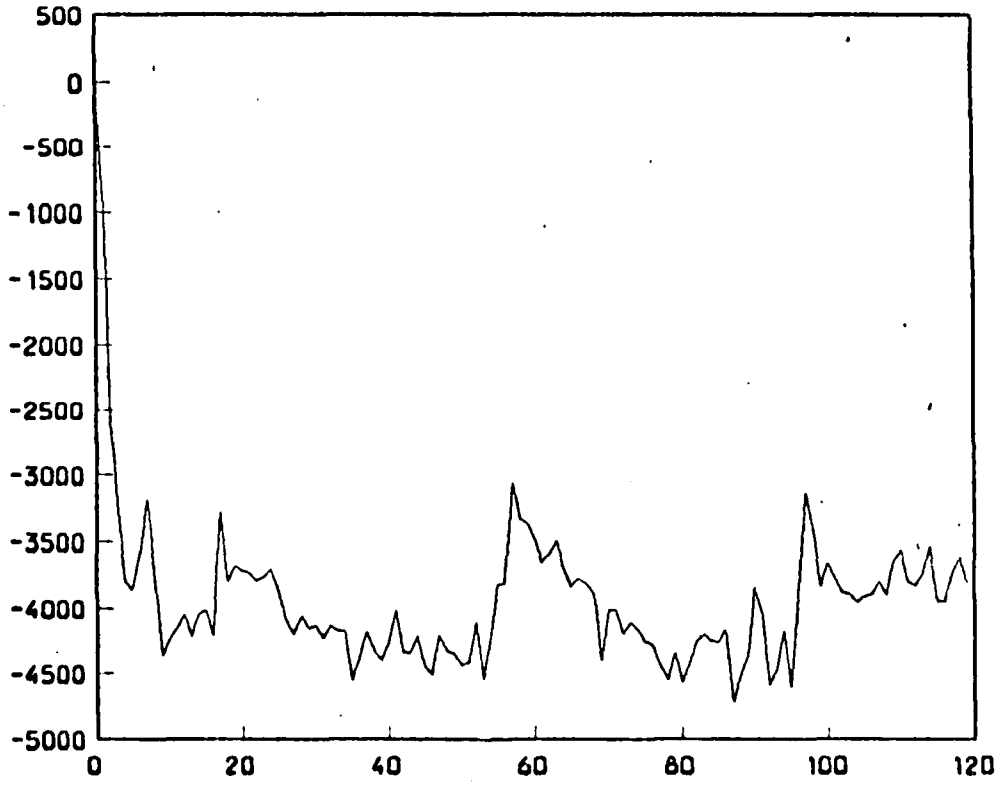
DAY 6, DIFFERENTIAL TEST, ROW 2, COLUMN 3



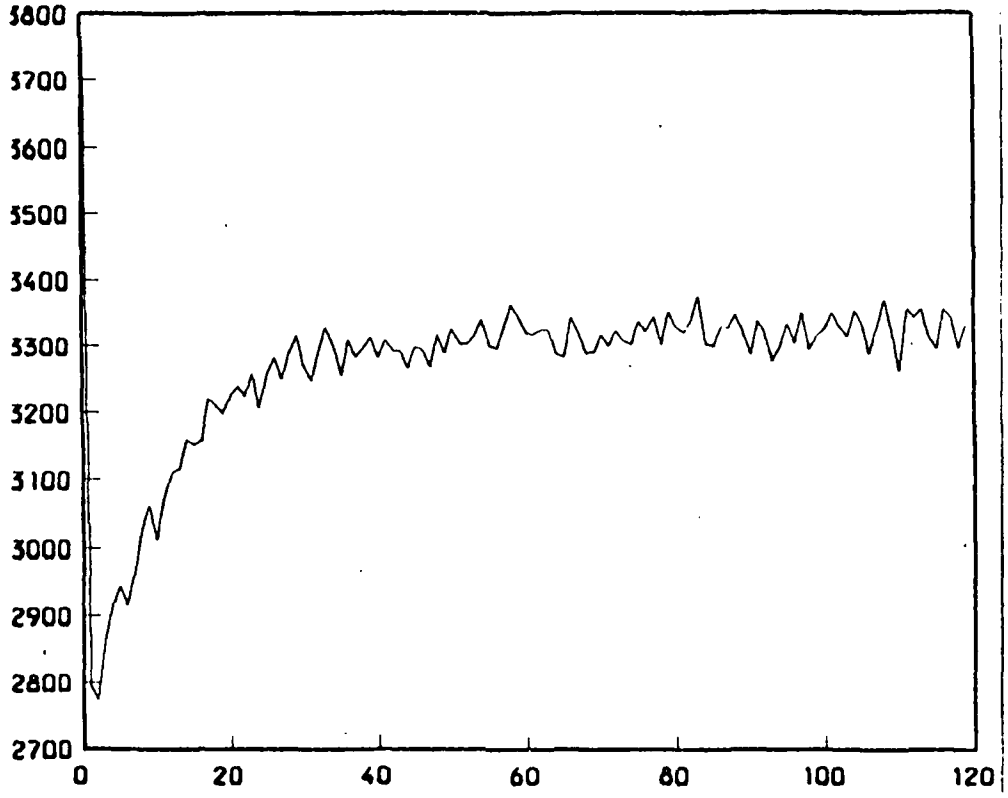
DAY 6, DIFFERENTIAL TEST, ROW 2, COLUMN 4



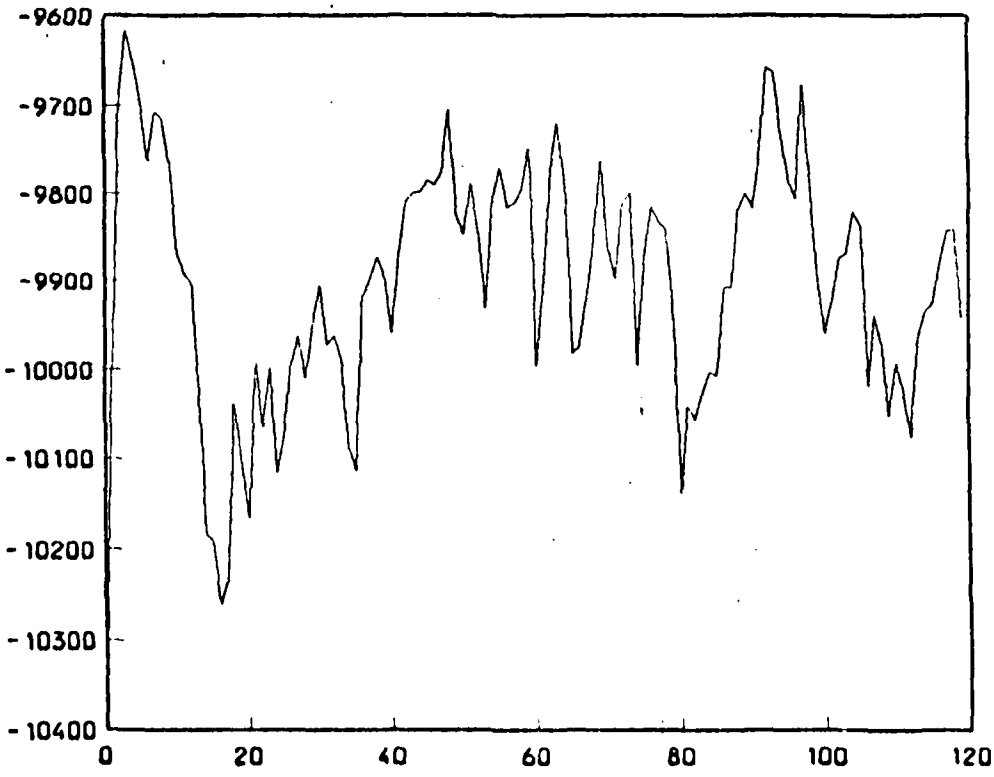
DAY 7, ABSOLUTE TEST, ROW 4, COLUMN 3



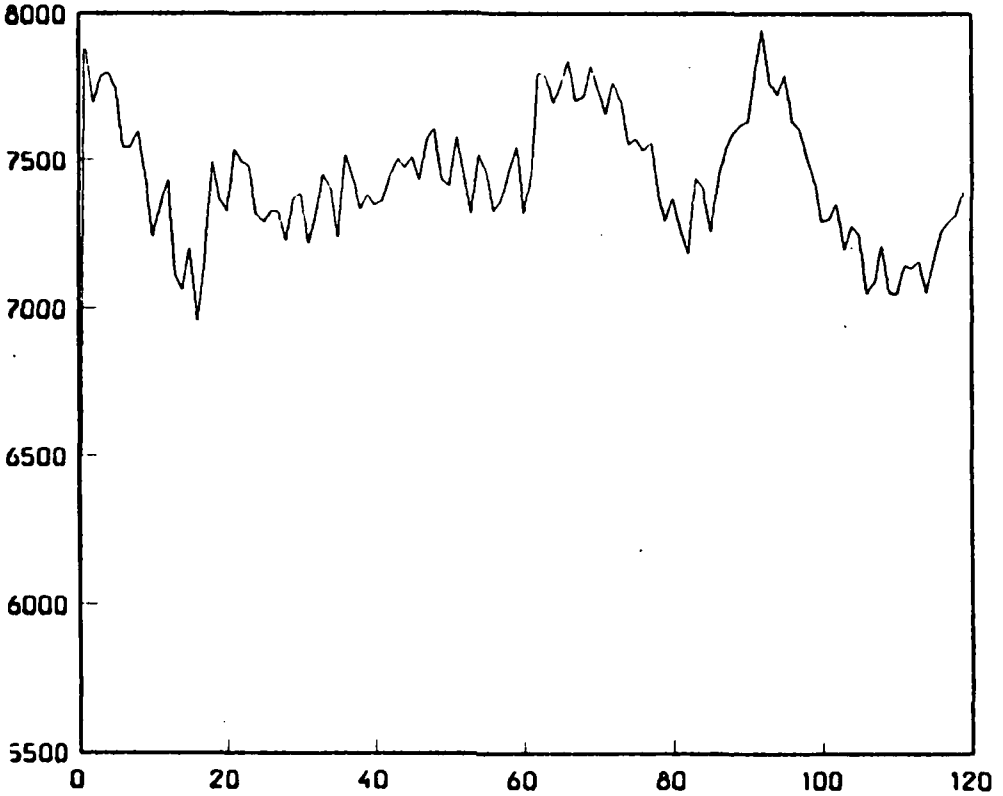
DAY 7, ABSOLUTE TEST, ROW 4, COLUMN 4



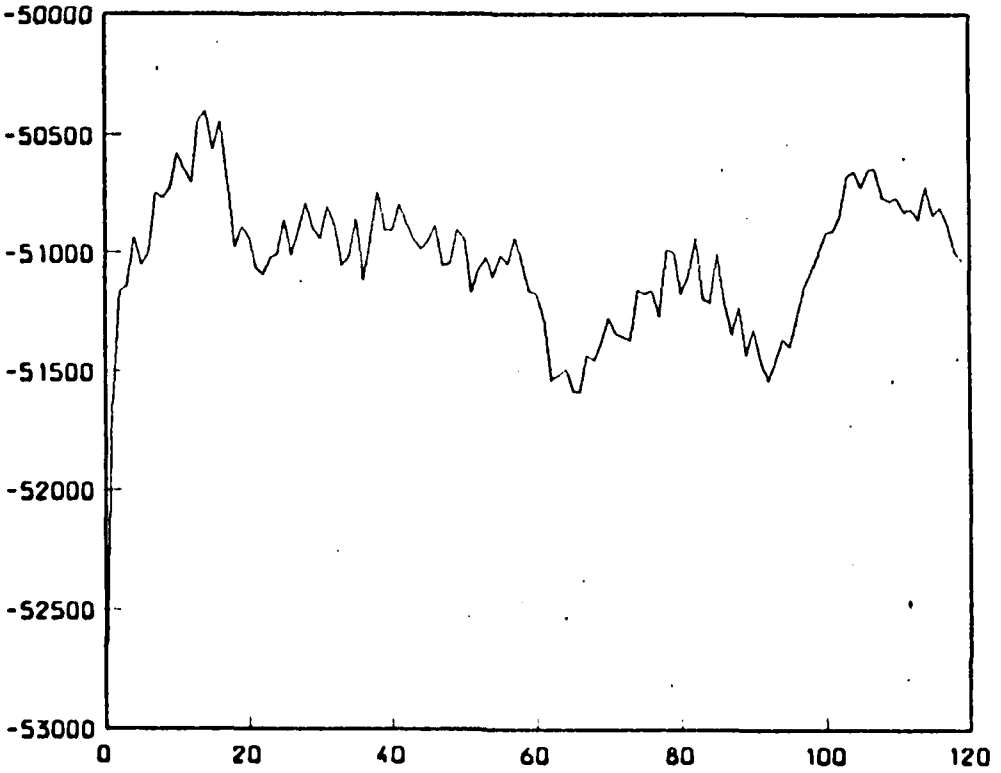
DAY 8, DIFFERENTIAL TEST, ROW 1, COLUMN 1



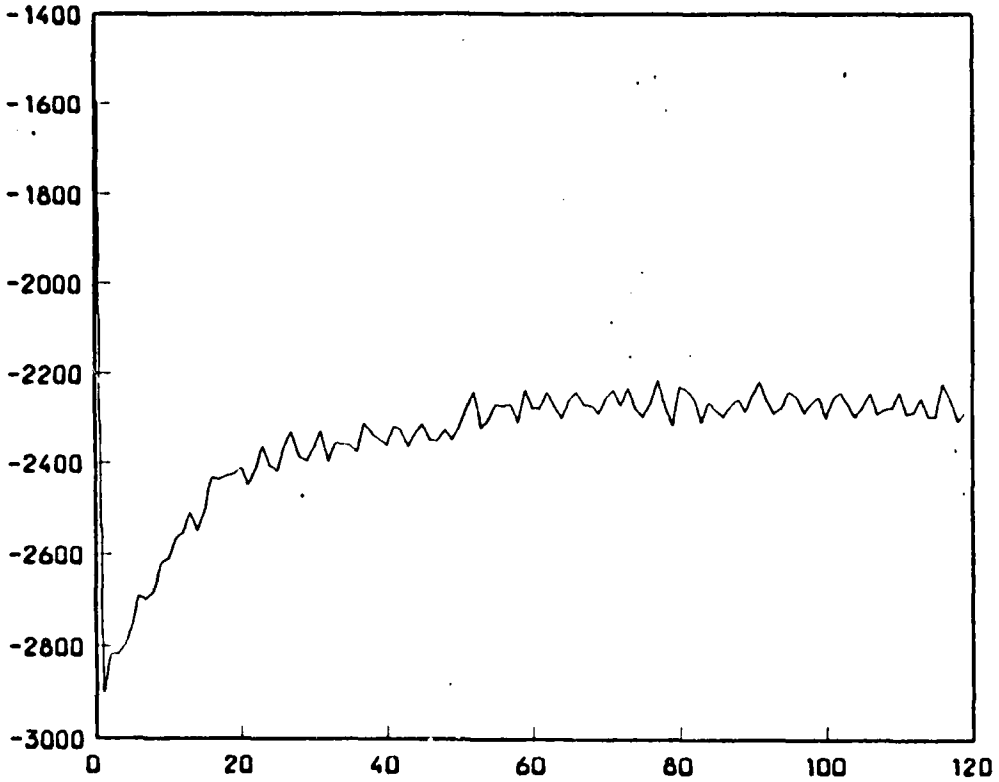
DAY 8, DIFFERENTIAL TEST, ROW 1, COLUMN 2



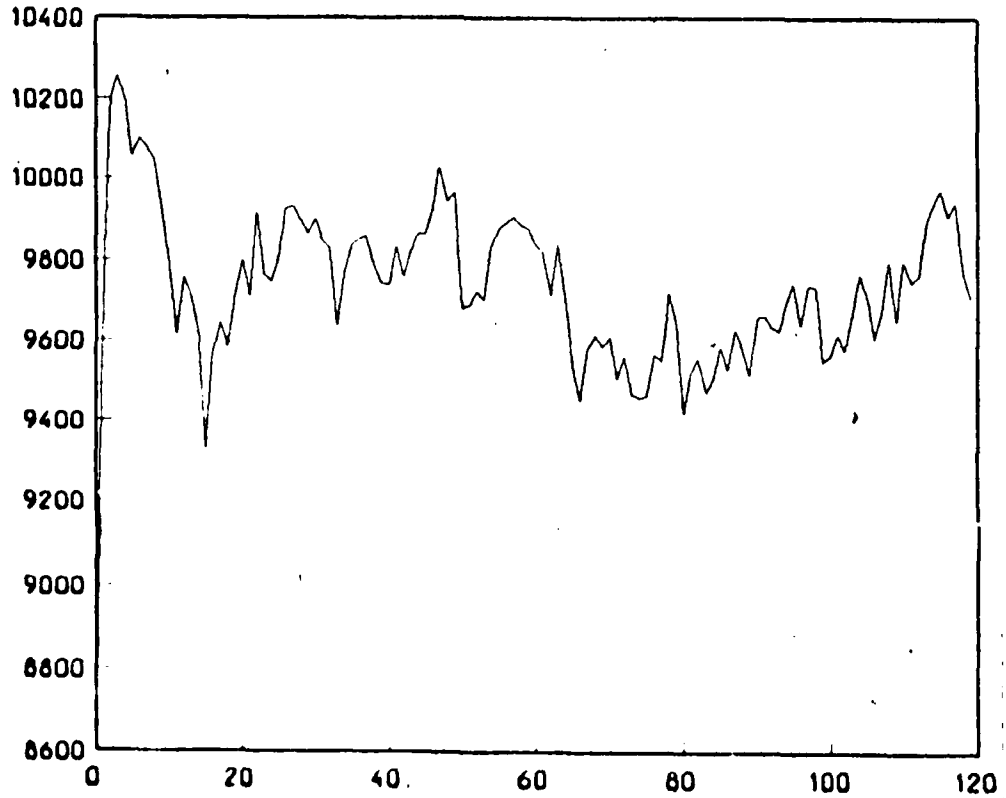
DAY 8, DIFFERENTIAL TEST, ROW 1, COLUMN 3



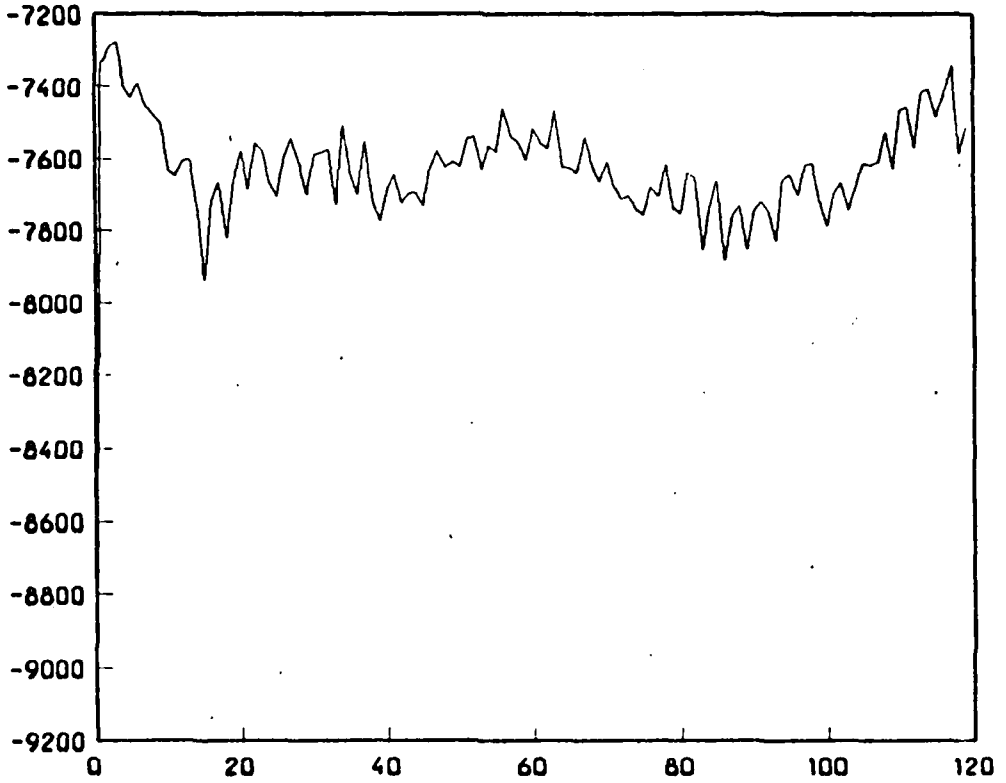
DAY 8, DIFFERENTIAL TEST, ROW 1, COLUMN 4



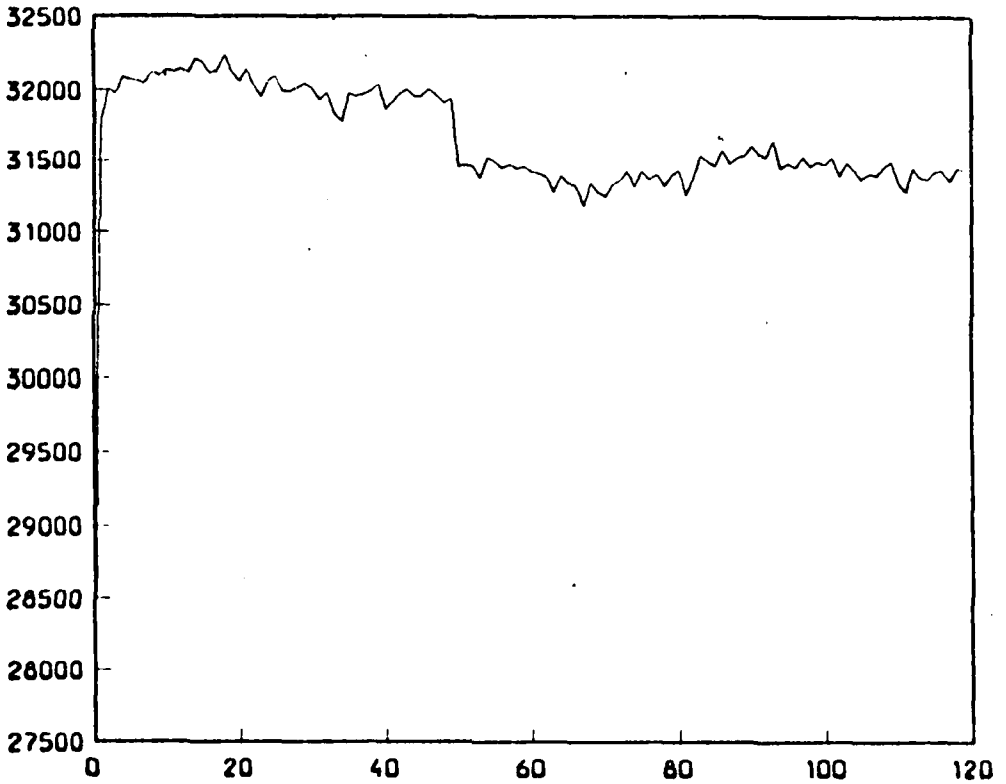
DAY 8, DIFFERENTIAL TEST, ROW 2, COLUMN 1



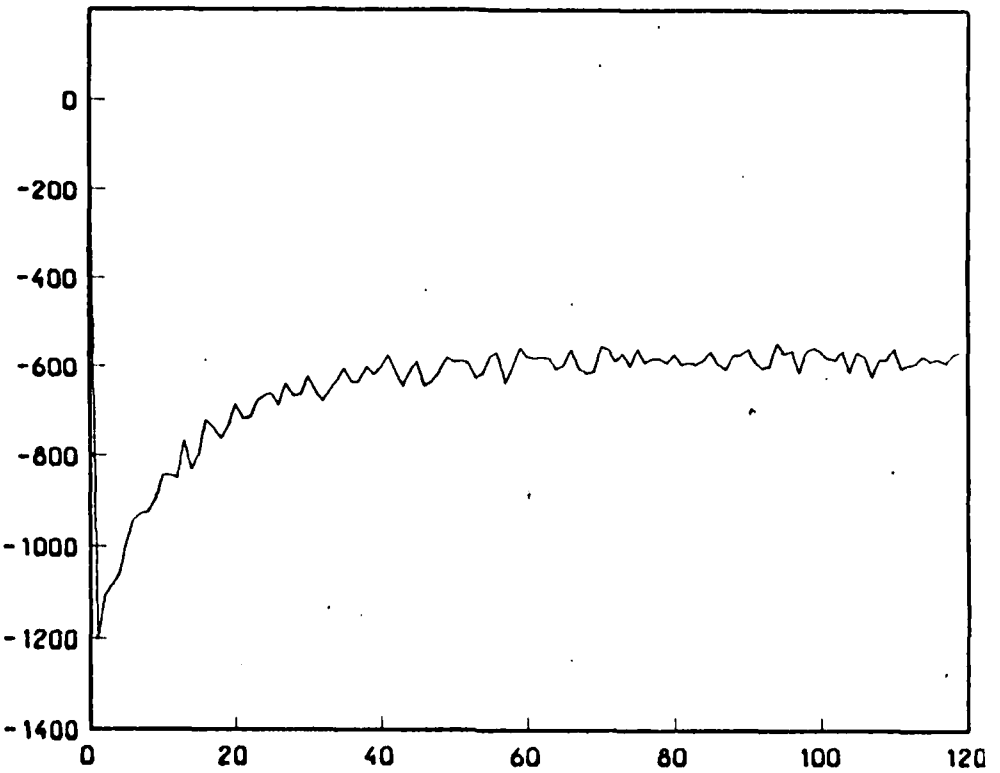
DAY 8, DIFFERENTIAL TEST, ROW 2, COLUMN 2



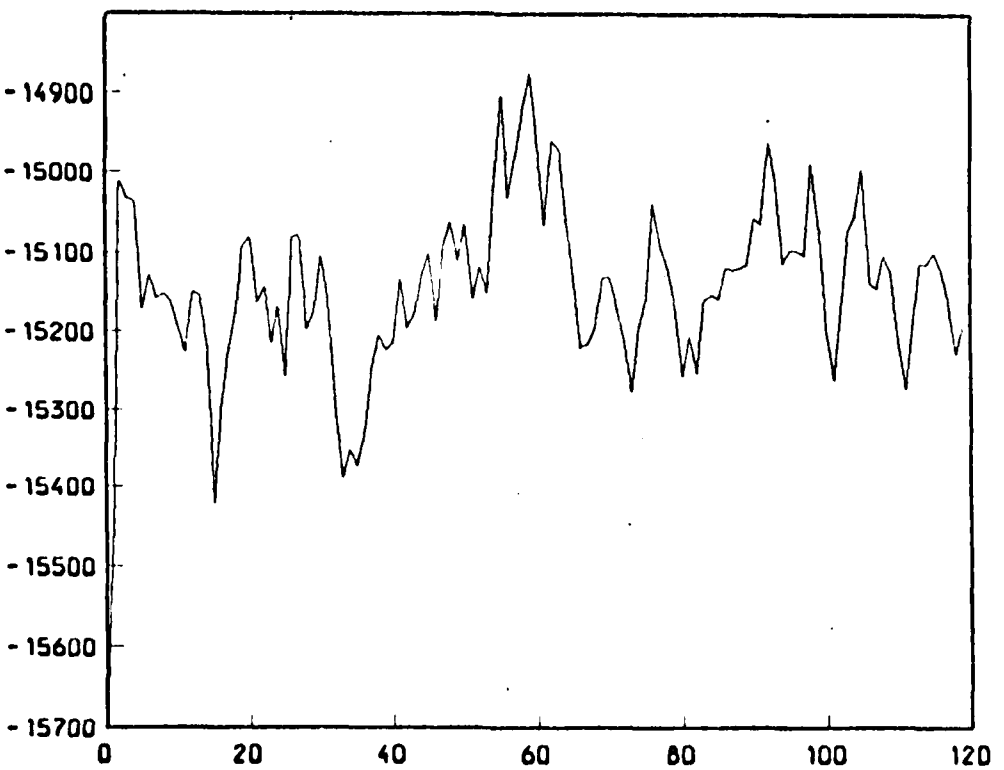
DAY 8, DIFFERENTIAL TEST, ROW 2, COLUMN 3



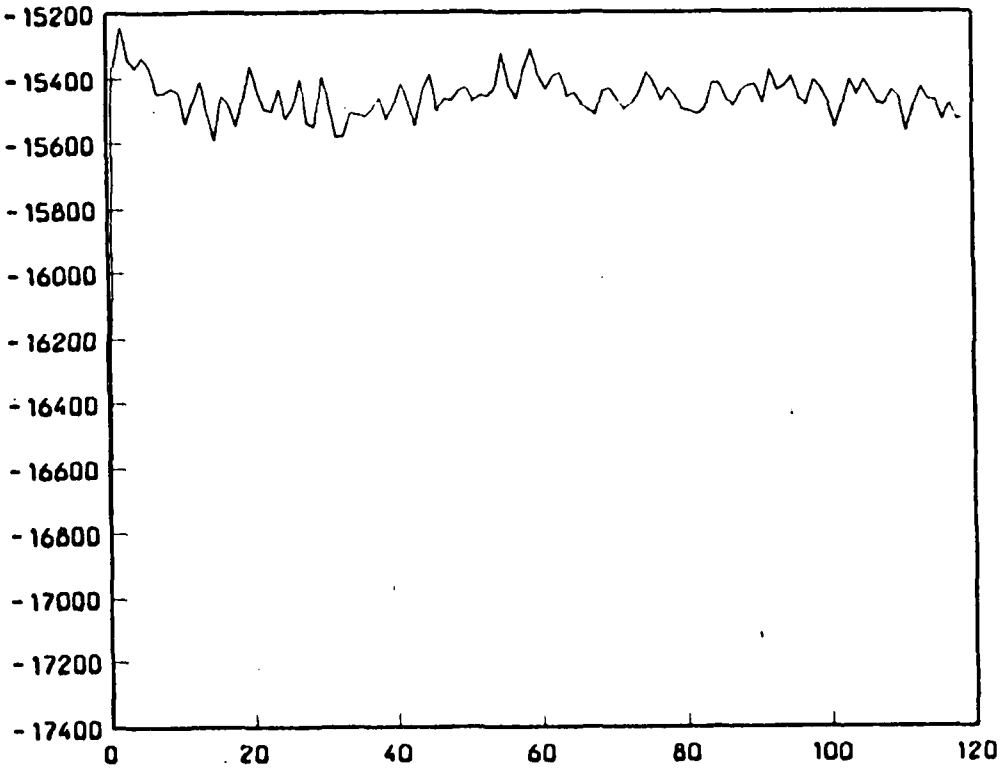
DAY 8, DIFFERENTIAL TEST, ROW 2, COLUMN 4



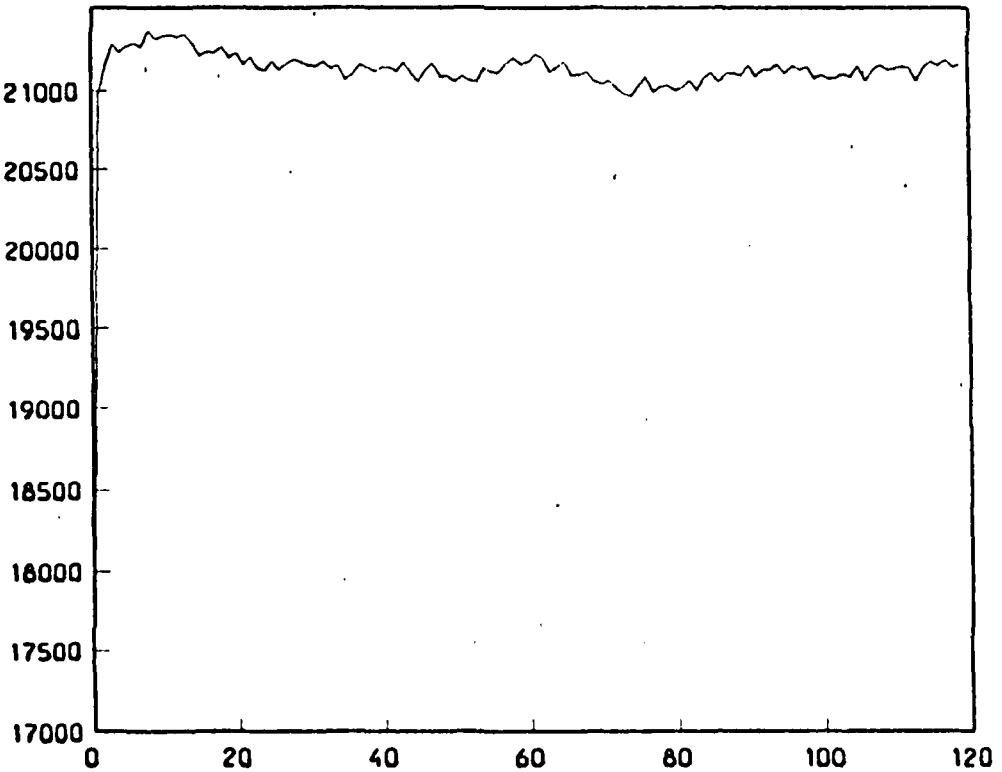
DAY 8, DIFFERENTIAL TEST, ROW 3, COLUMN 1



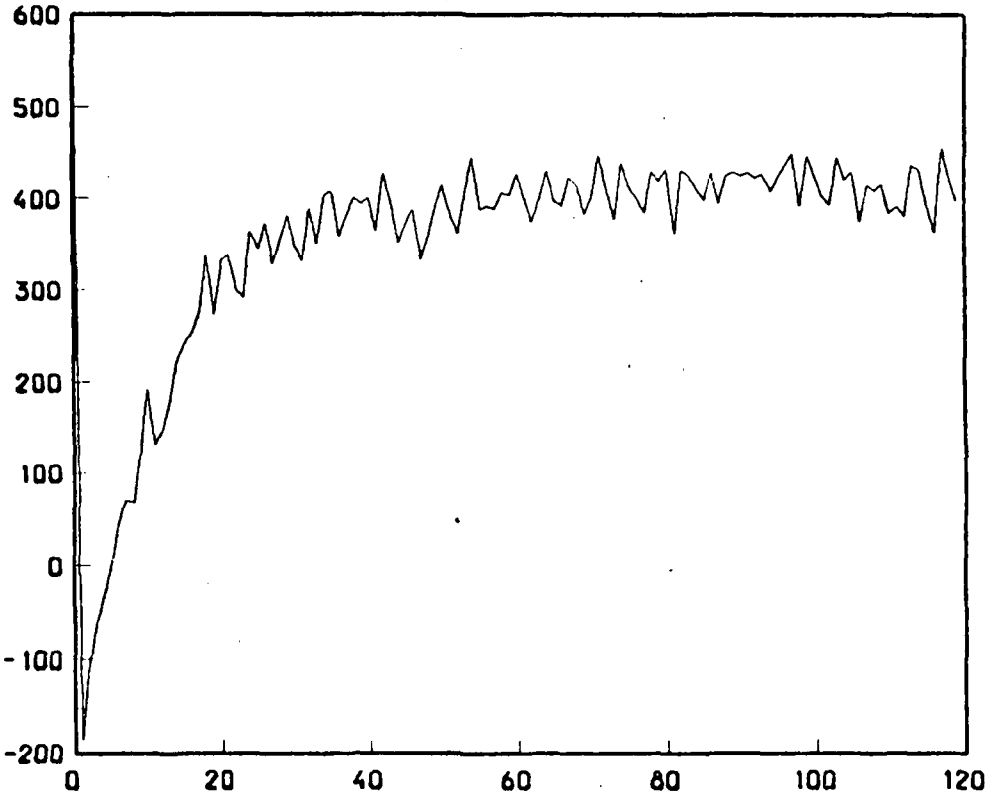
DAY 8, DIFFERENTIAL TEST, ROW 3, COLUMN 2



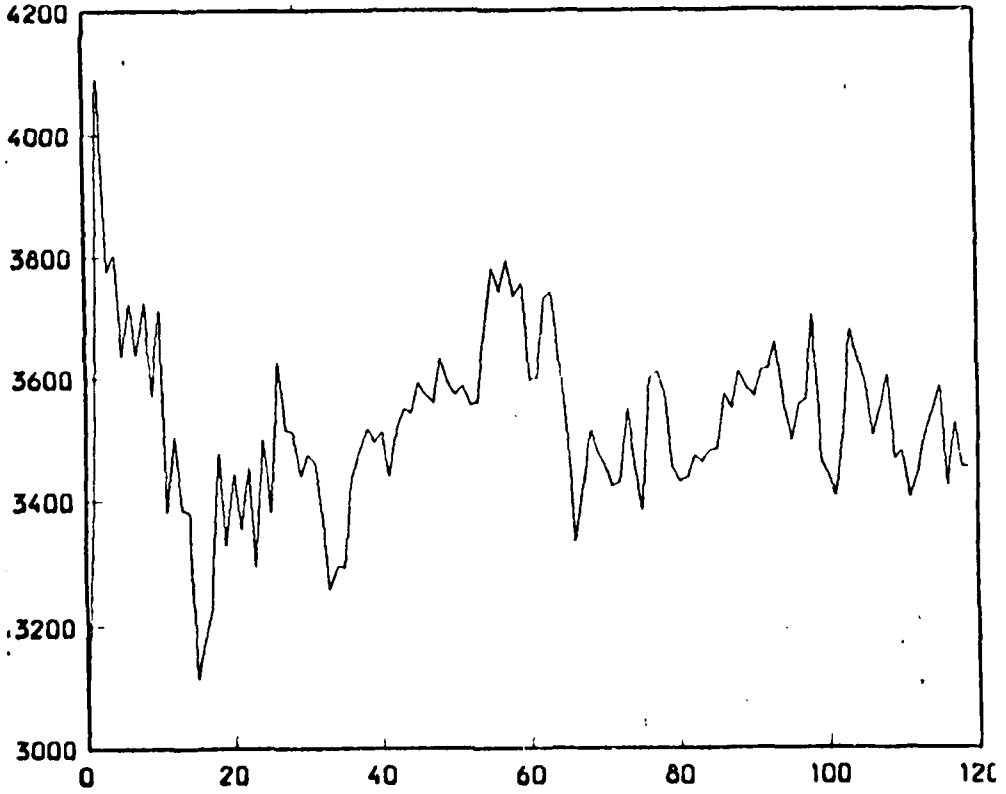
DAY 8, DIFFERENTIAL TEST, ROW 3, COLUMN 3



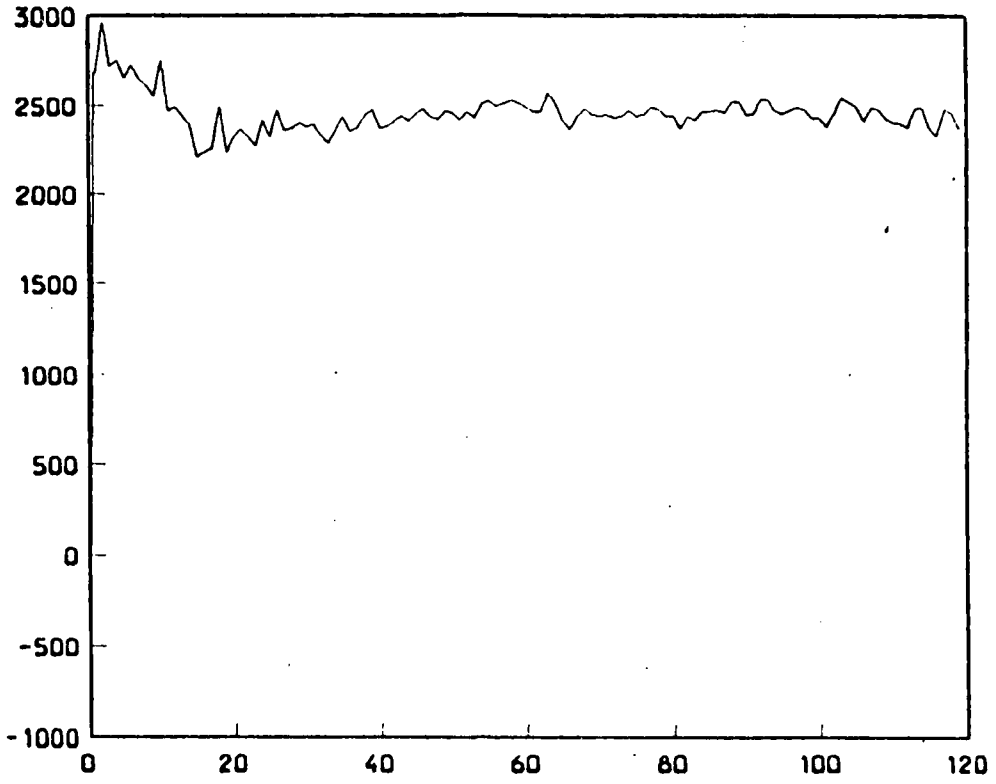
DAY 8, DIFFERENTIAL TEST, ROW 3, COLUMN 4



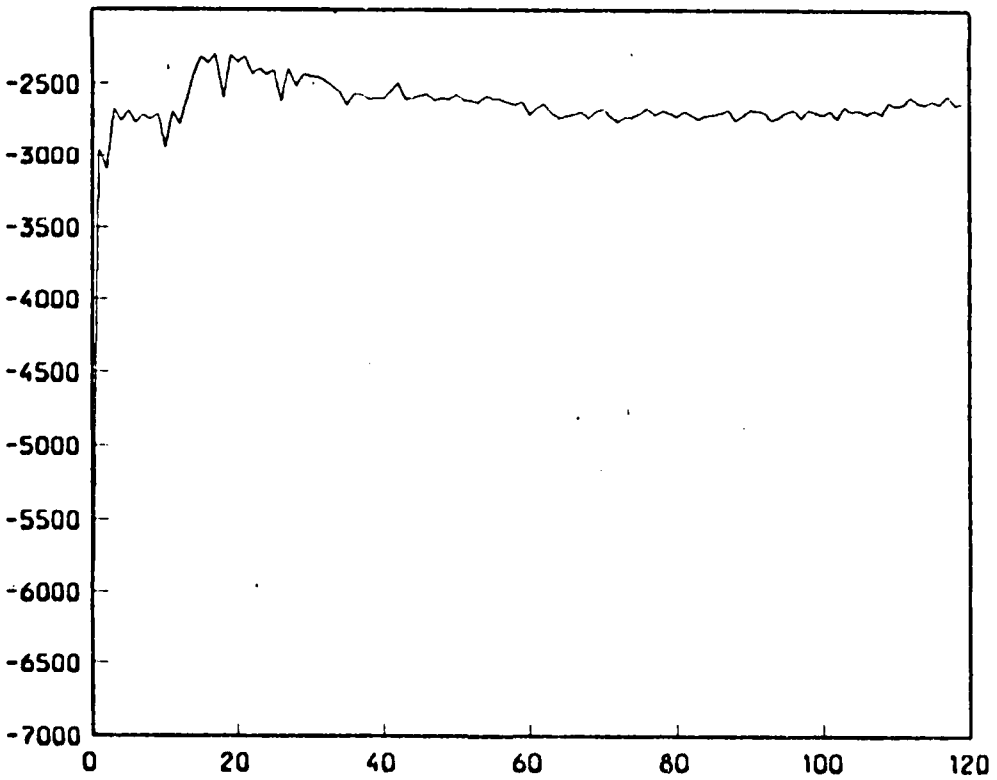
DAY 8, DIFFERENTIAL TEST, ROW 4, COLUMN 1



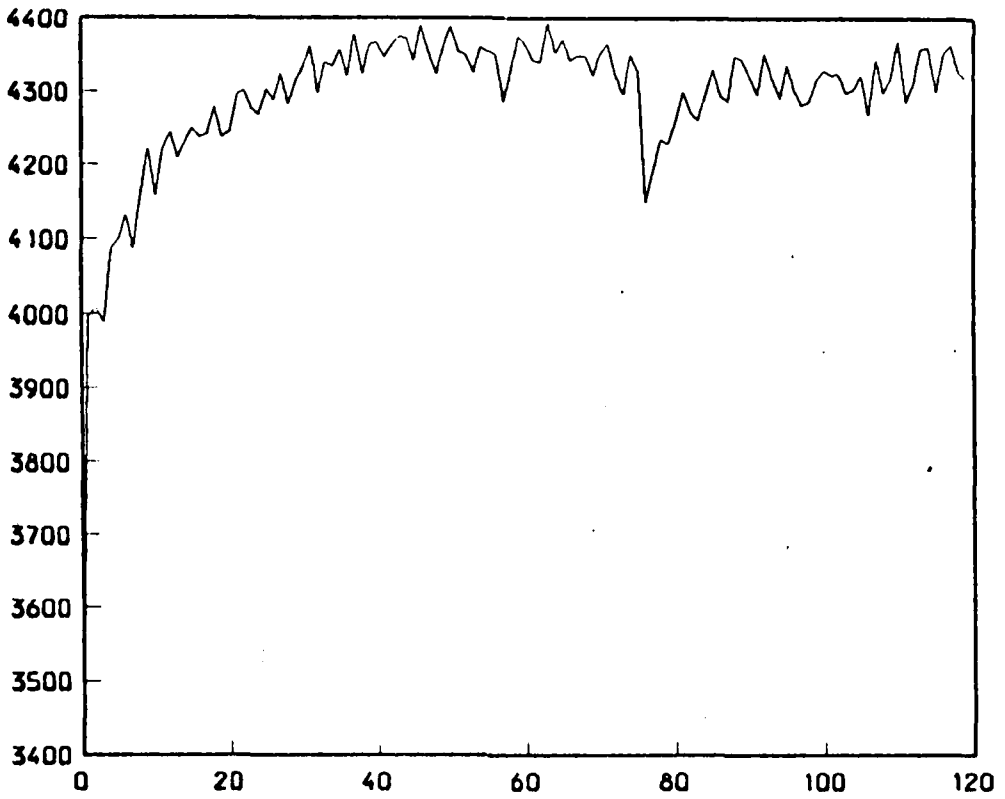
DAY 8, DIFFERENTIAL TEST, ROW 4, COLUMN 2



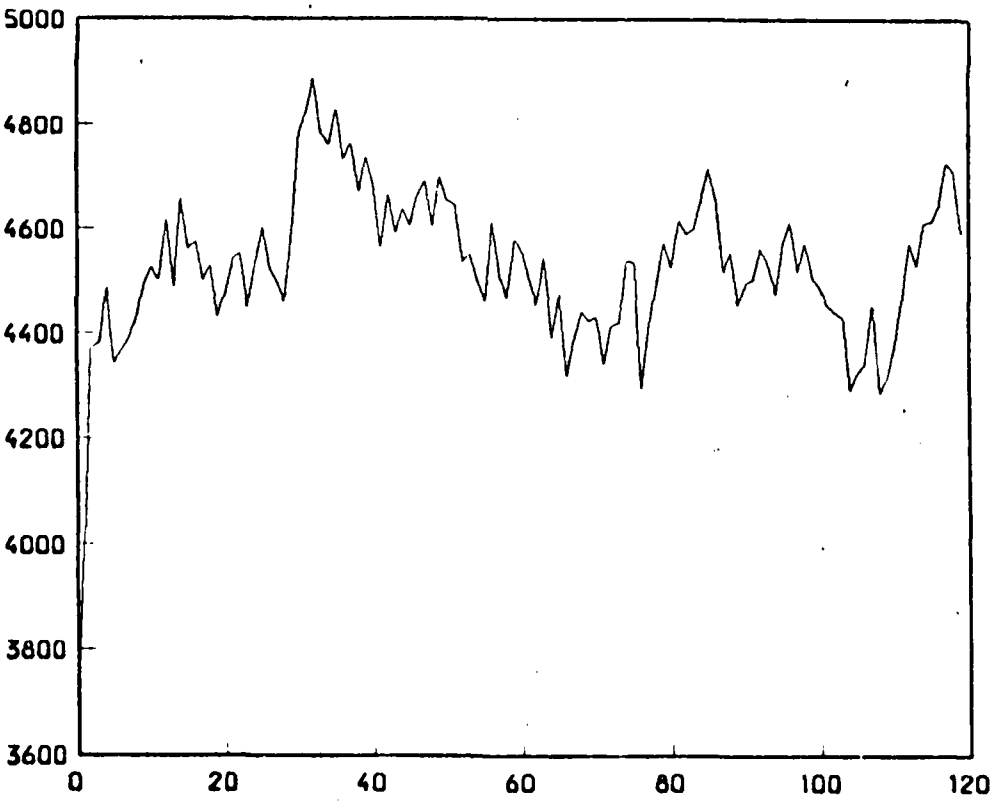
DAY 8, DIFFERENTIAL TEST, ROW 4, COLUMN 3



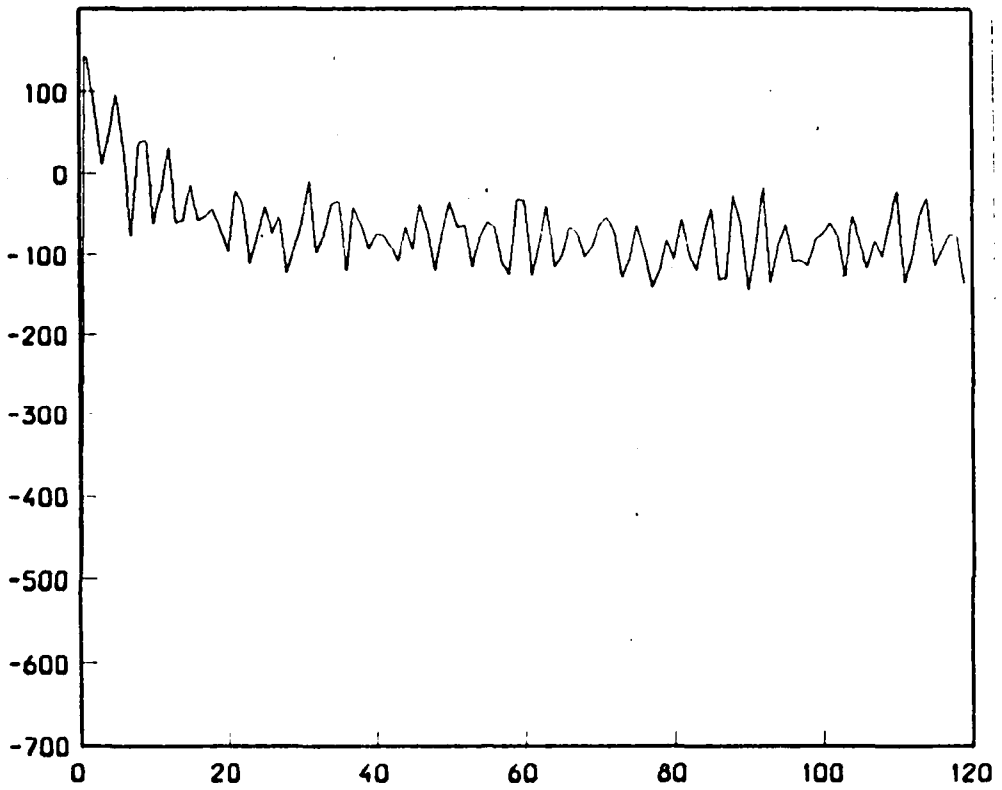
DAY 8, DIFFERENTIAL TEST, ROW 4, COLUMN 4



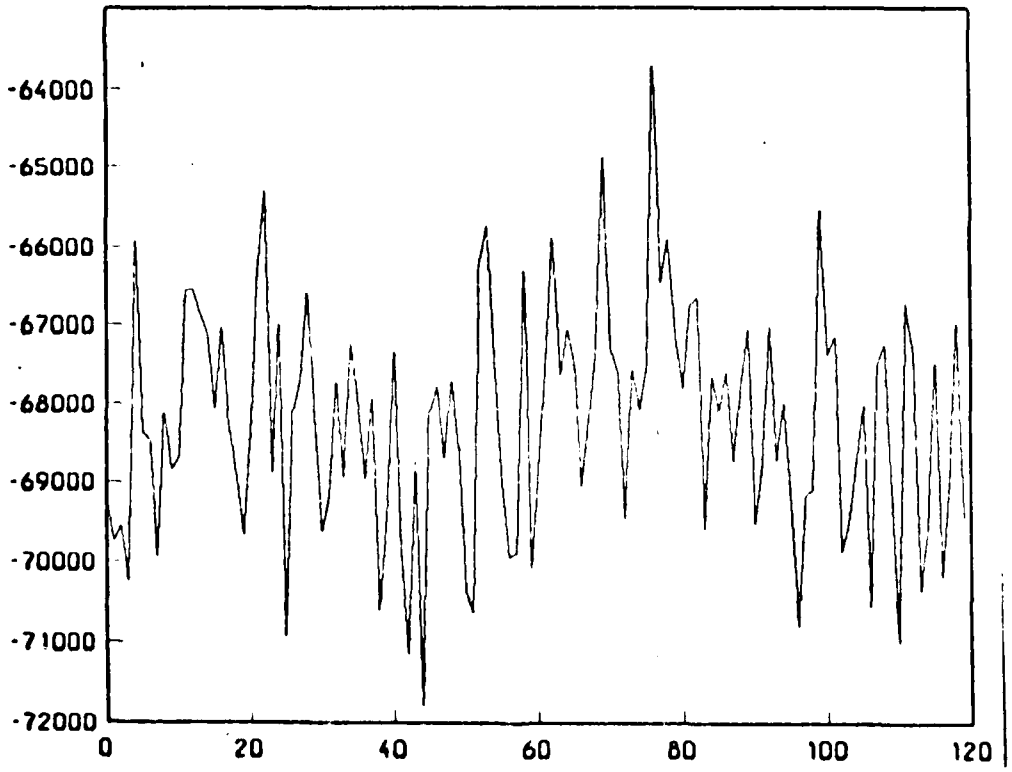
DAY 8, ABSOLUTE TEST, ROW 1, COLUMN 1



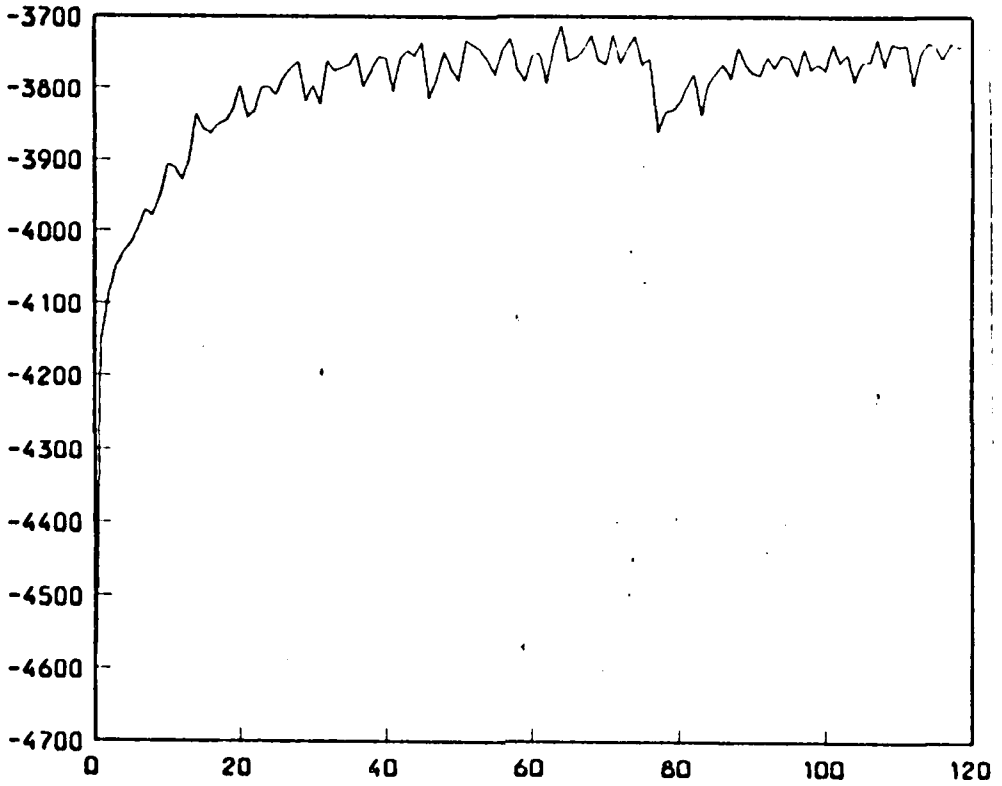
DAY 8, ABSOLUTE TEST, ROW 1, COLUMN 2



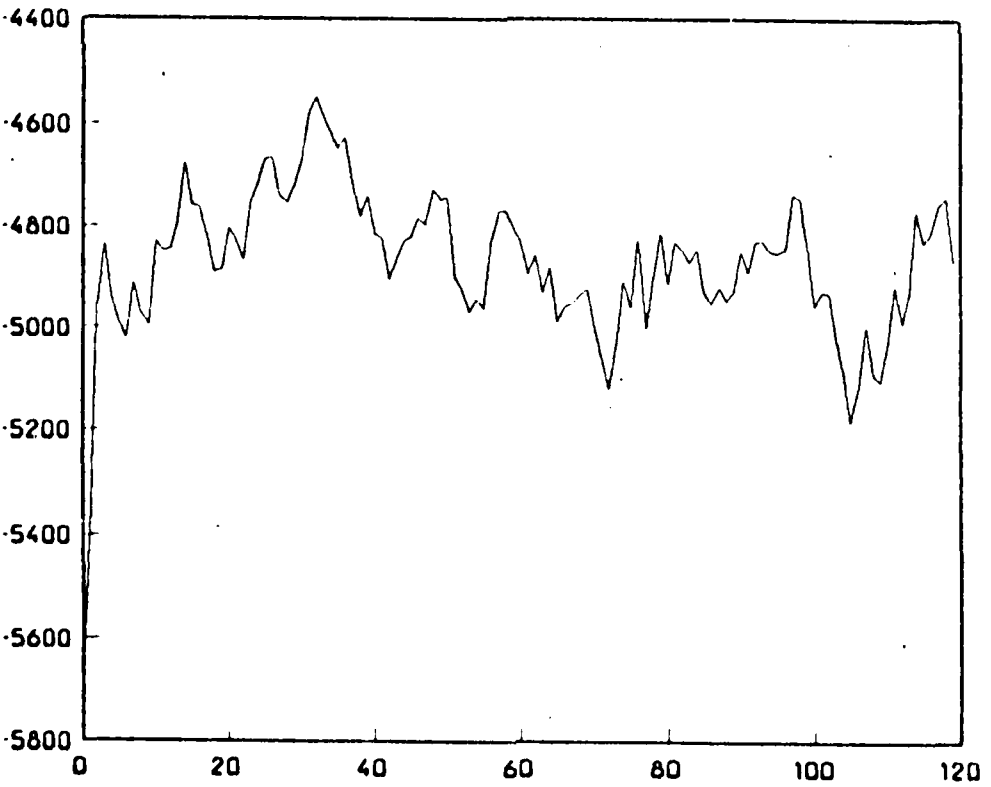
DAY 8, ABSOLUTE TEST, ROW 1, COLUMN 3



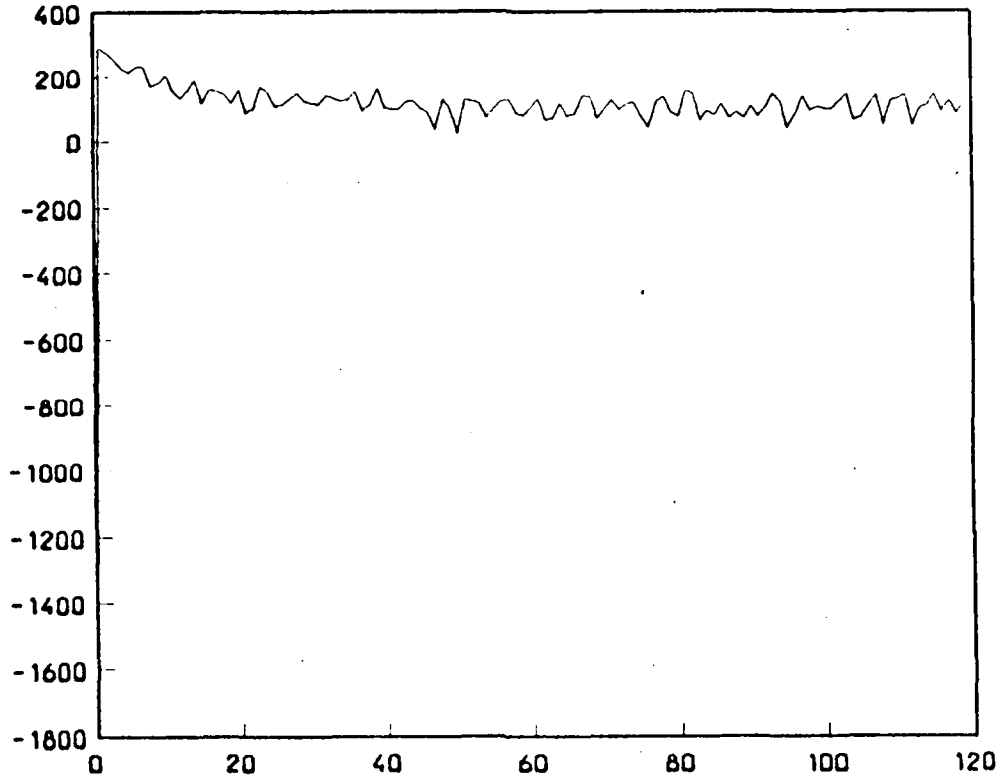
DAY 8, ABSOLUTE TEST, ROW 1, COLUMN 4



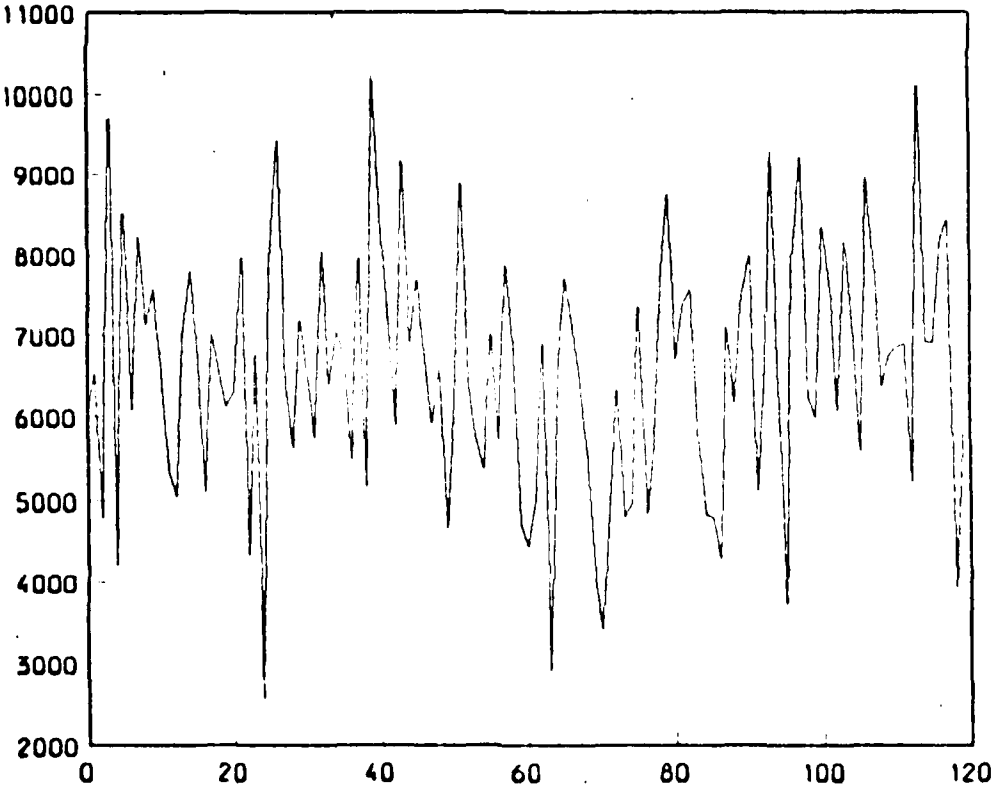
DAY 8, ABSOLUTE TEST, ROW 2, COLUMN 1



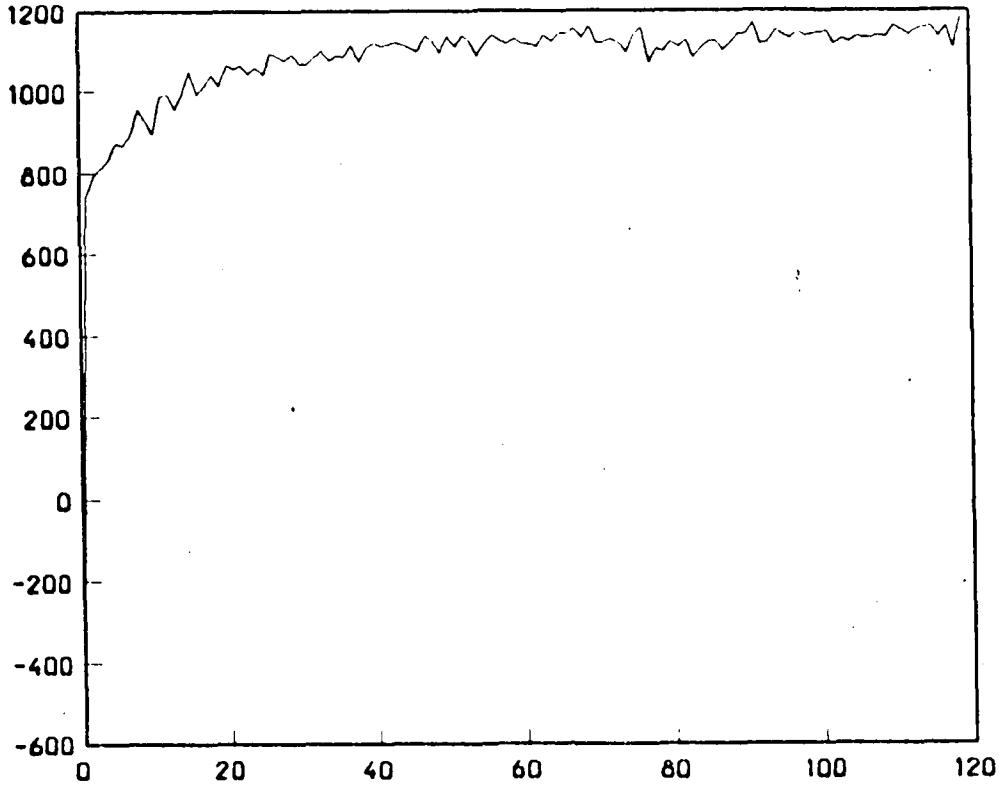
DAY 8, ABSOLUTE TEST, ROW 2, COLUMN 2



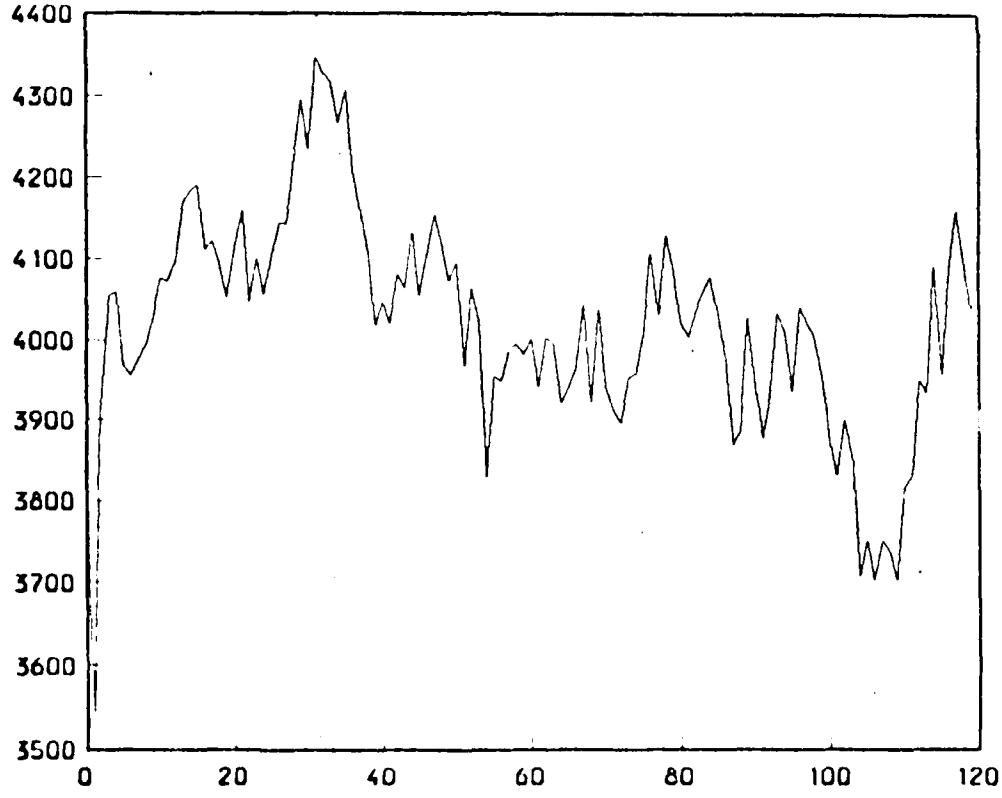
DAY 8, ABSOLUTE TEST, ROW 2, COLUMN 3



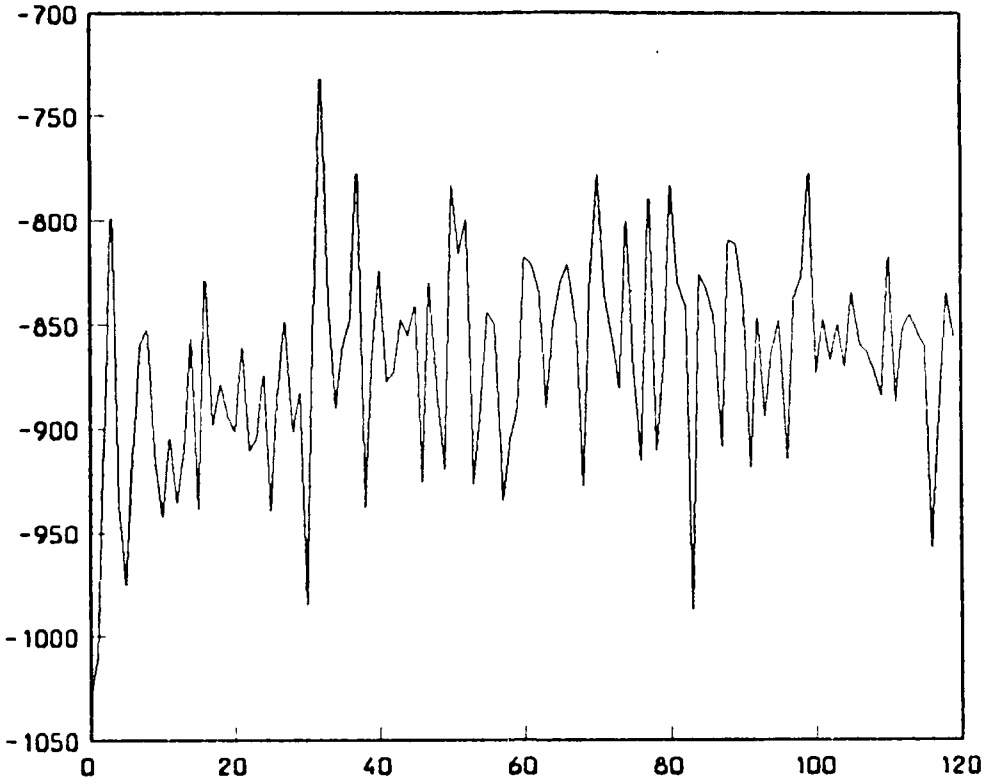
DAY 8, ABSOLUTE TEST, ROW 2, COLUMN 4



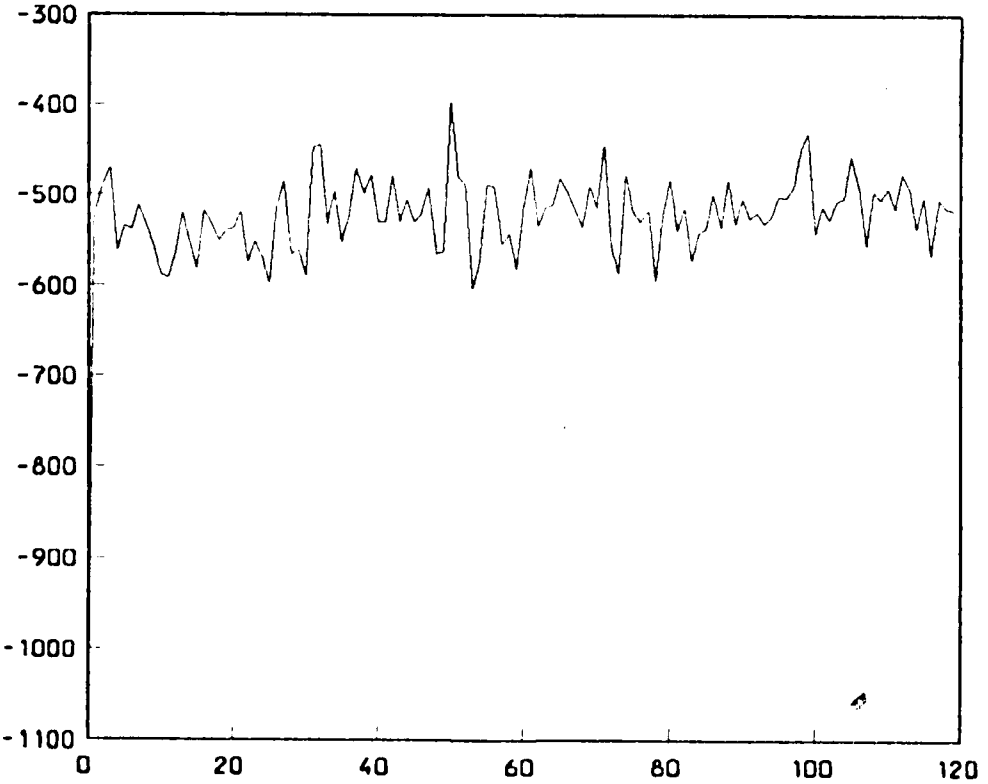
DAY 8, ABSOLUTE TEST, ROW 3, COLUMN 1



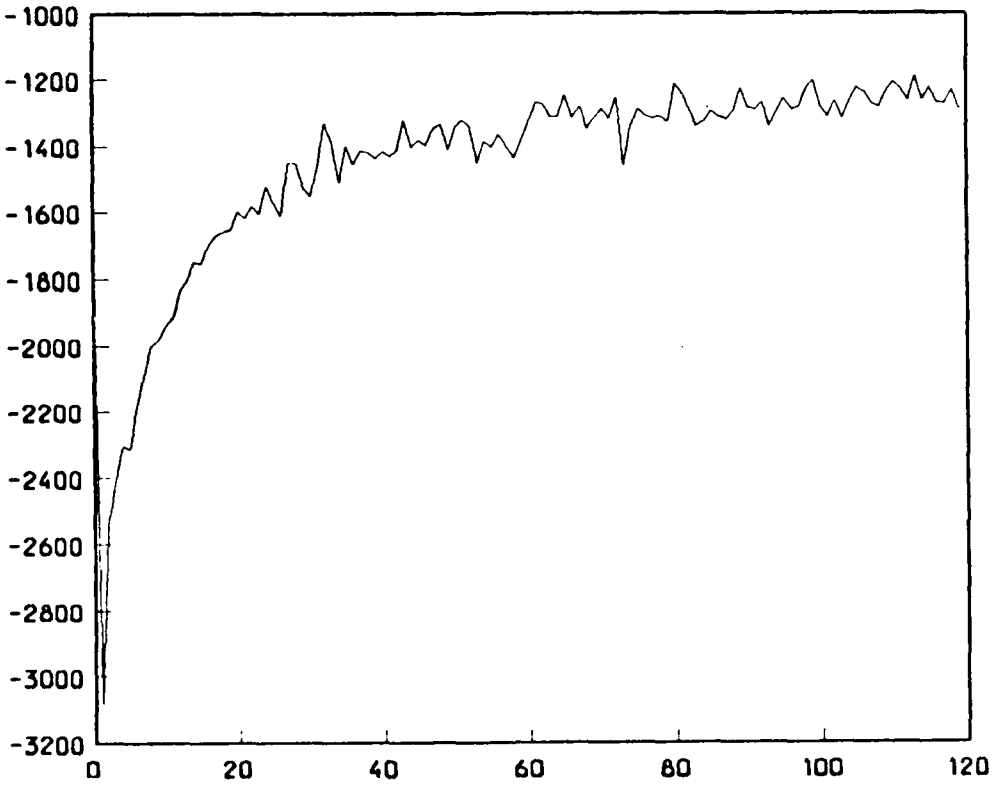
DAY 8, ABSOLUTE TEST, ROW 3, COLUMN 2



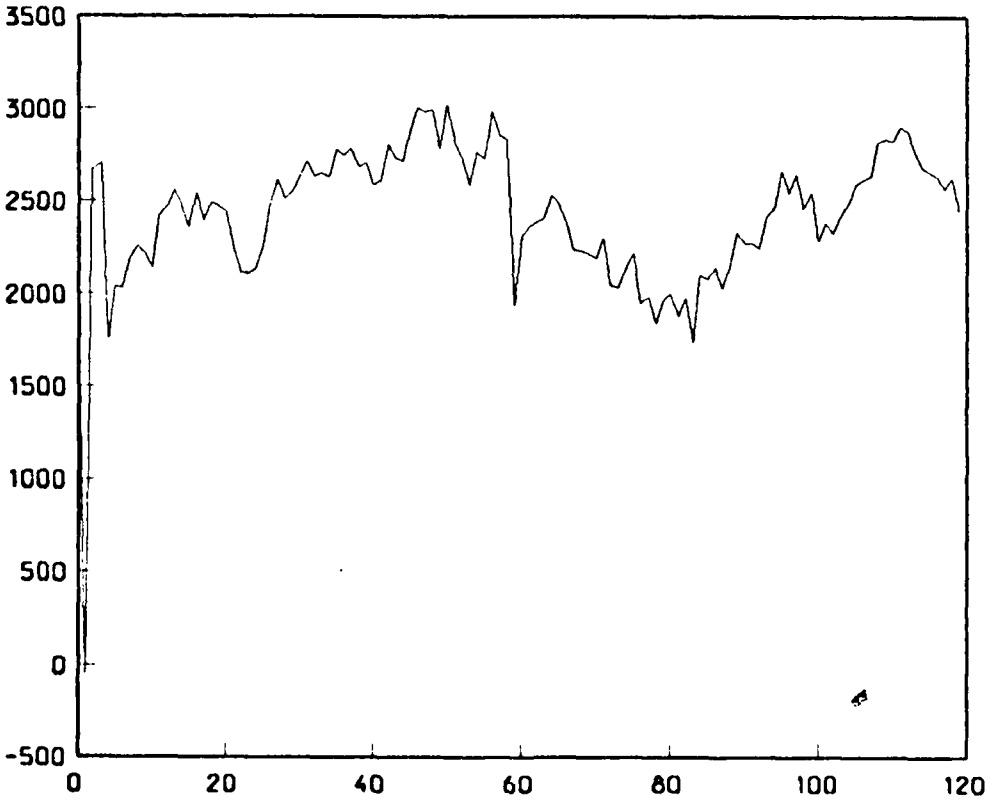
DAY 11, ABSOLUTE TEST, ROW 1, COLUMN 3



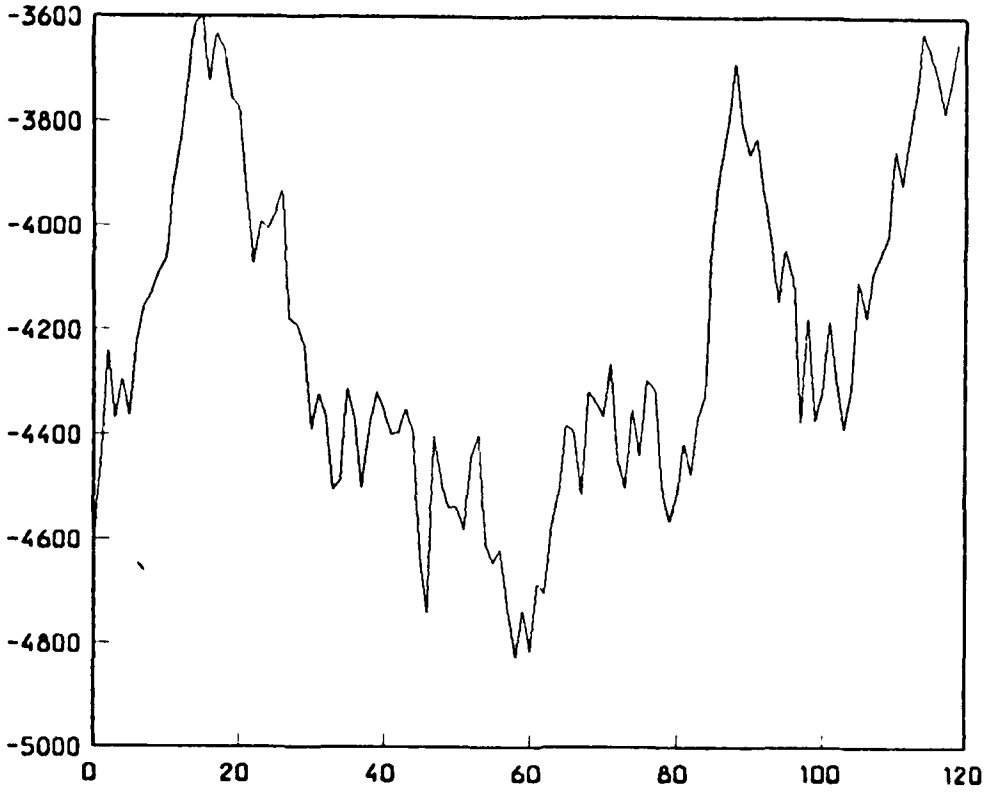
DAY 11, ABSOLUTE TEST, ROW 1, COLUMN 4



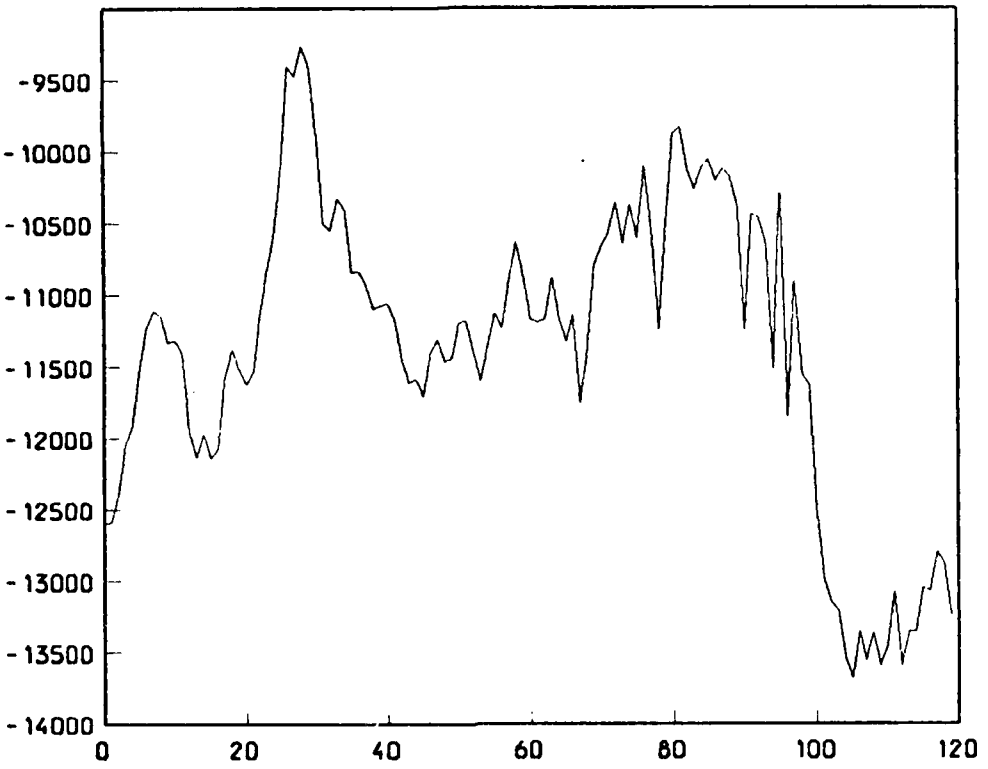
DAY 11, ABSOLUTE TEST, ROW 1, COLUMN 1



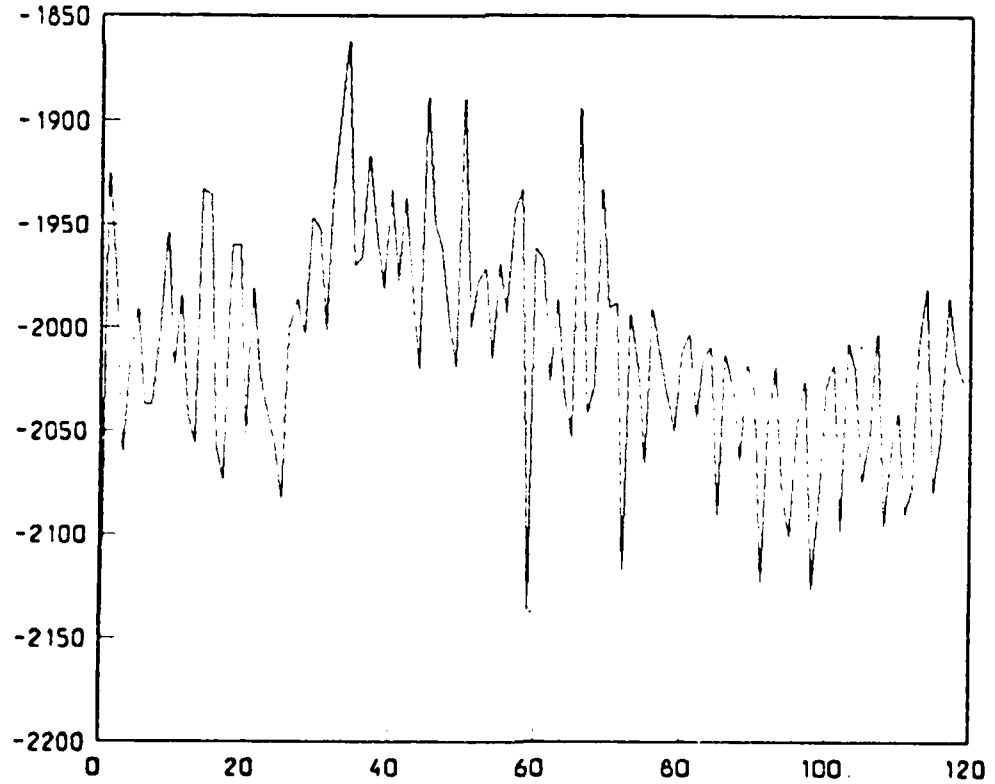
DAY 11, ABSOLUTE TEST, ROW 1, COLUMN 2



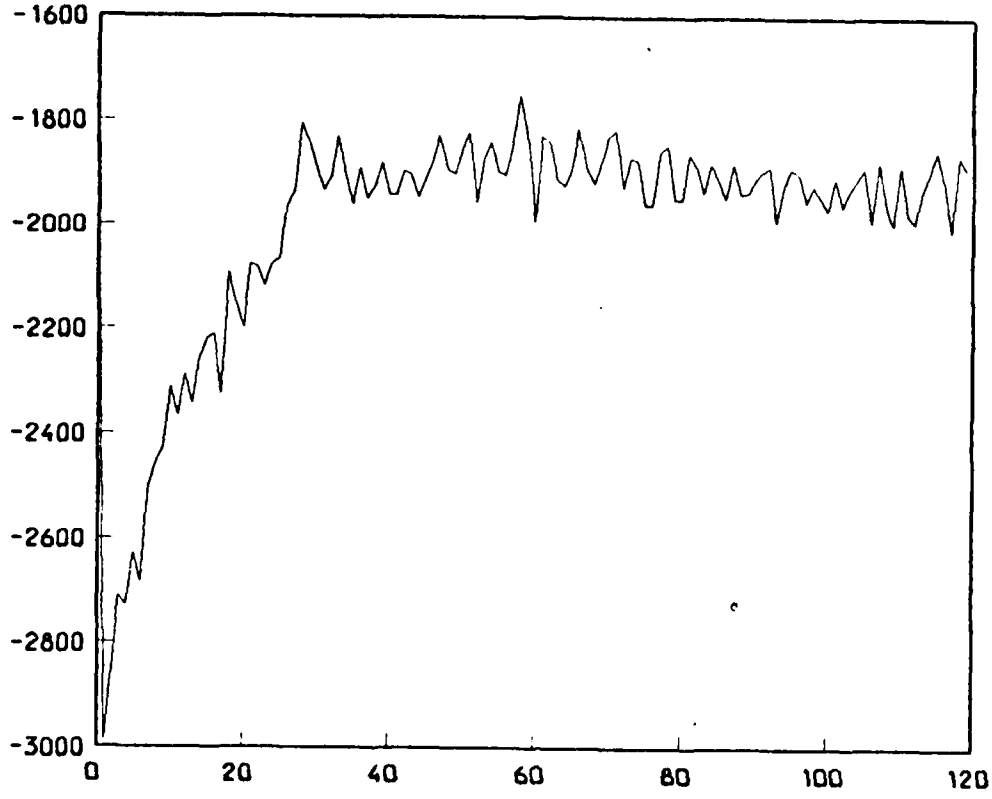
DAY 11, ELECTRODE #5



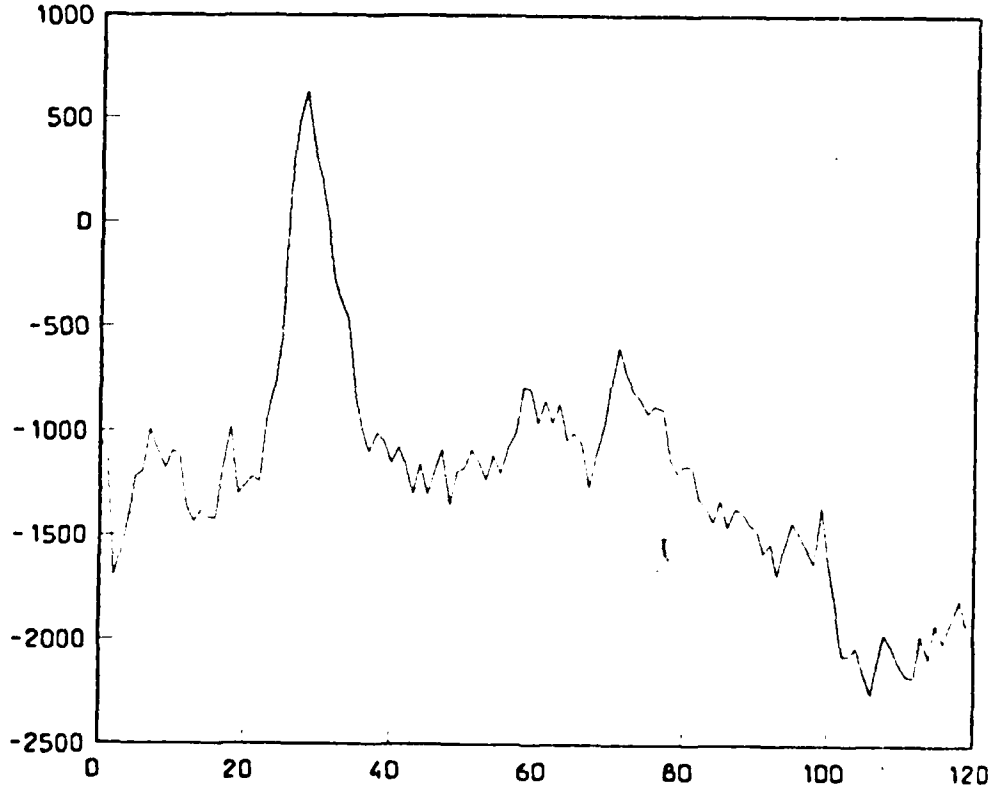
DAY 11, DIFFERENTIAL TEST, ROW 4, COLUMN 3



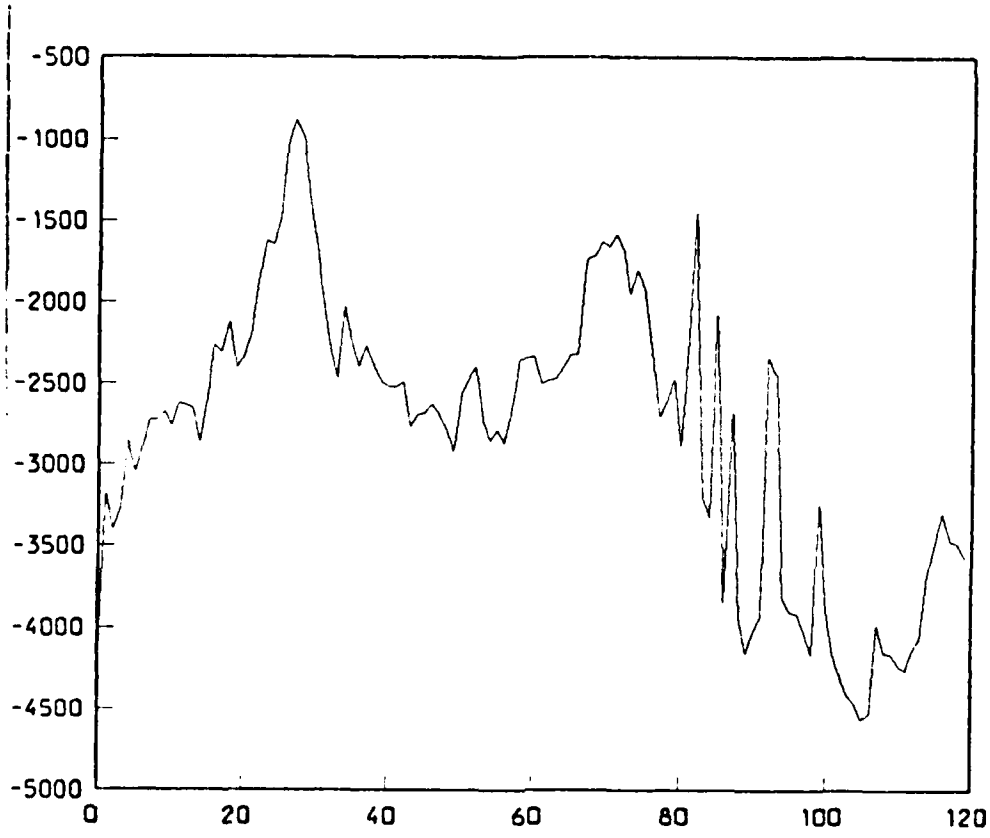
DAY 11, DIFFERENTIAL TEST, ROW 4, COLUMN 4



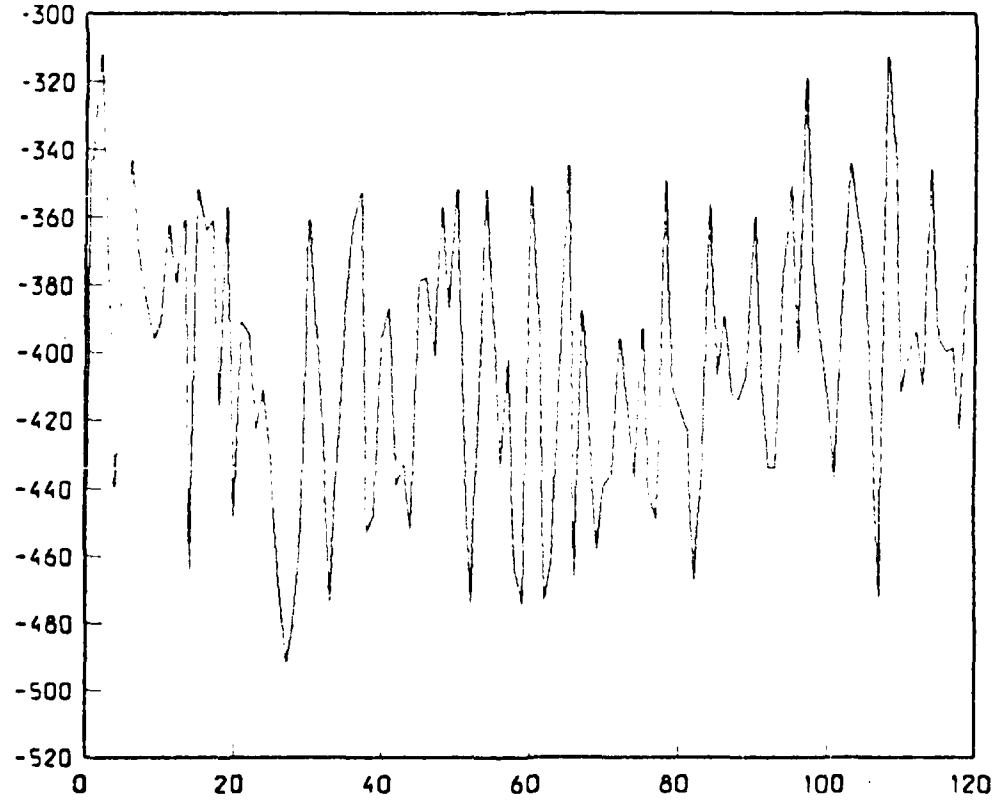
DAY 11, DIFFERENTIAL TEST, ROW 4, COLUMN 1



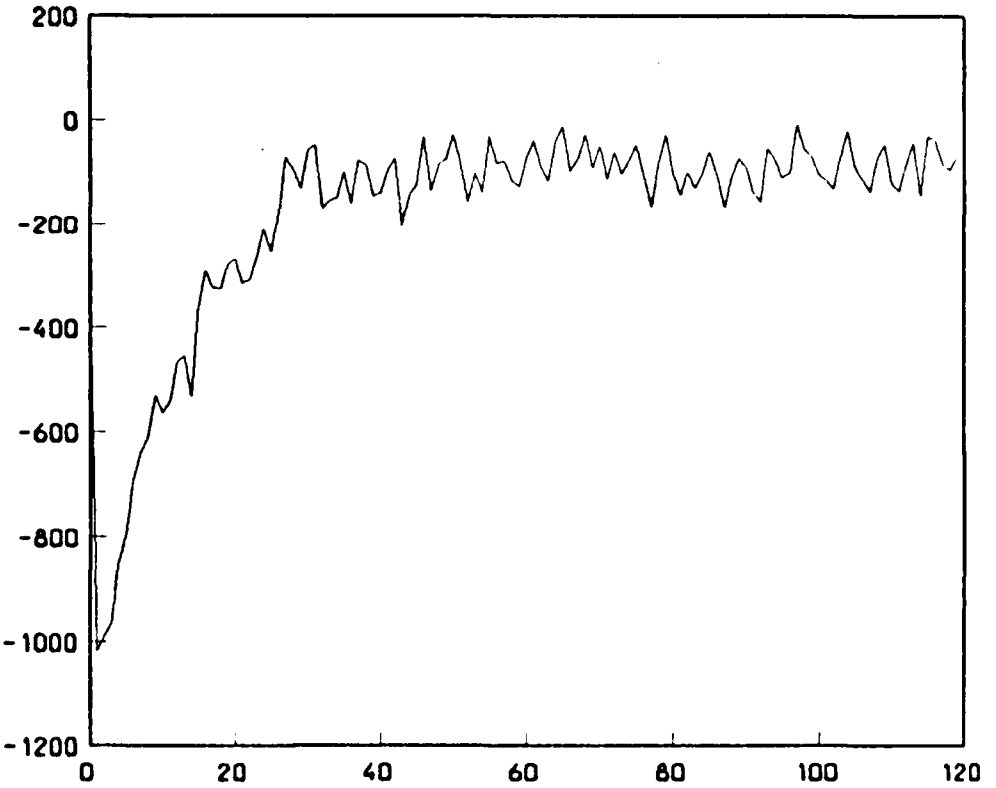
DAY 11, DIFFERENTIAL TEST, ROW 4, COLUMN 2



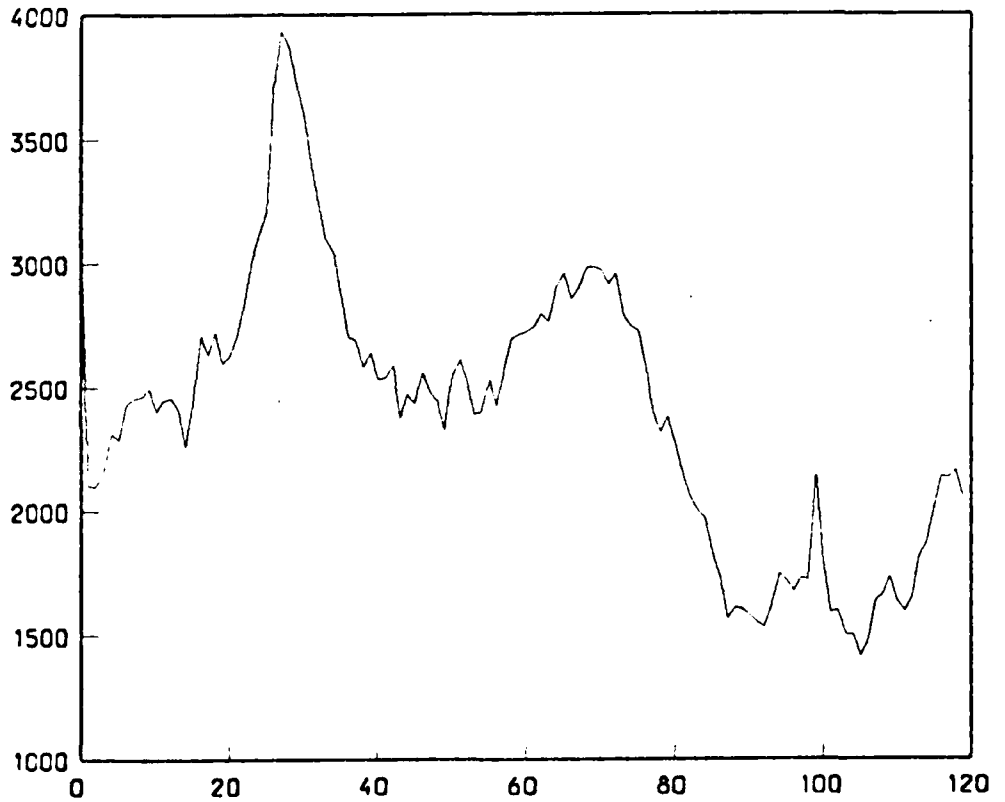
DAY 11, DIFFERENTIAL TEST, ROW 3, COLUMN 3



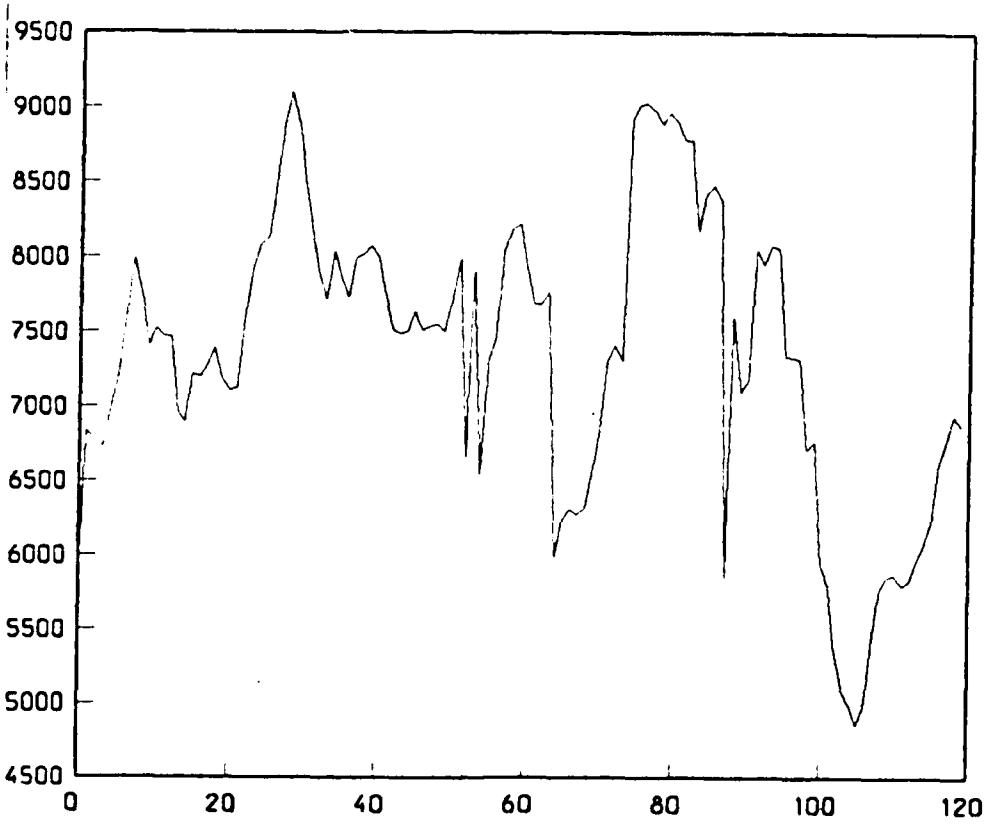
DAY 11, DIFFERENTIAL TEST, ROW 3, COLUMN 4



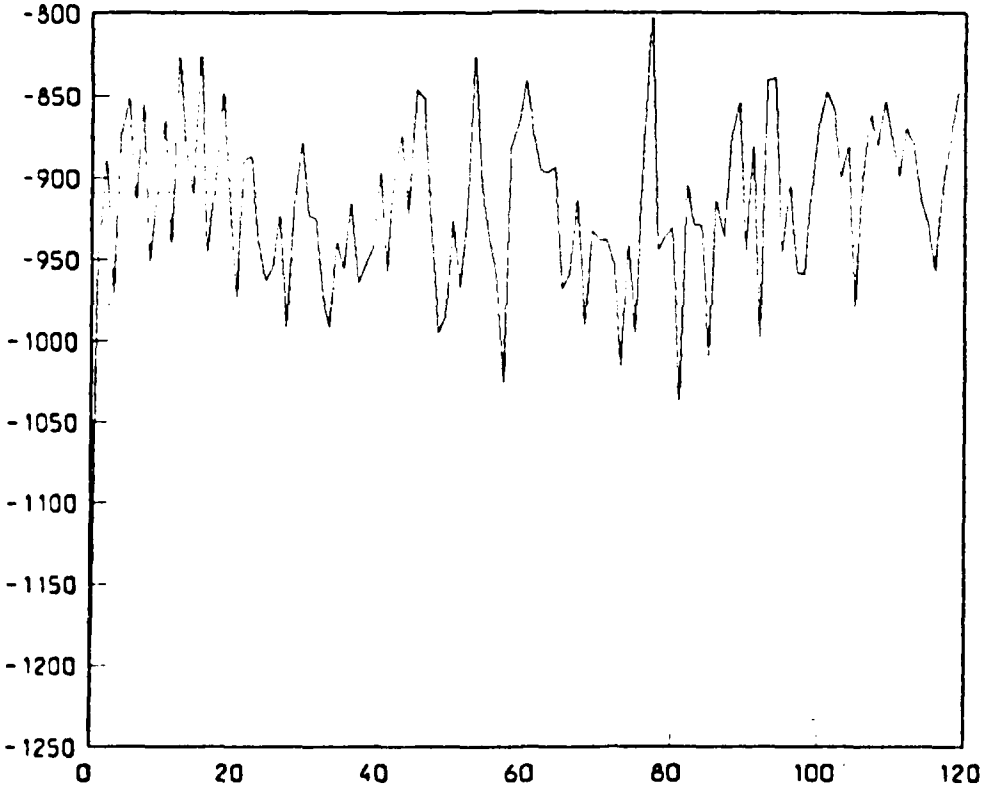
DAY 11, DIFFERENTIAL TEST, ROW 3, COLUMN 1



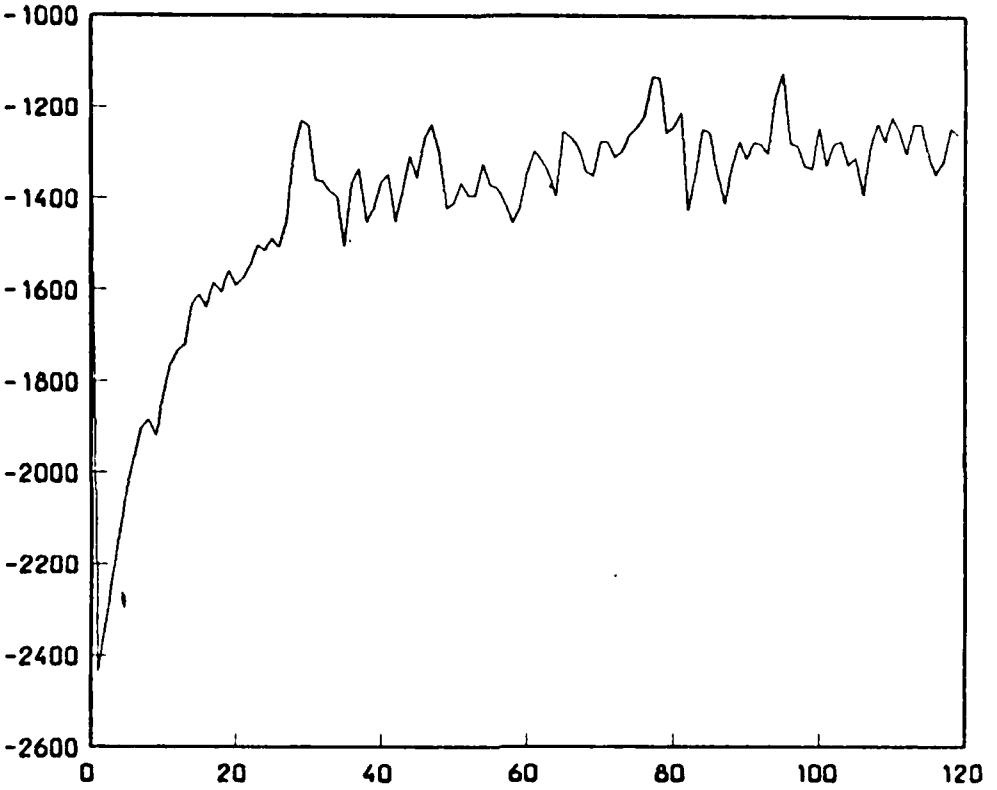
DAY 11, DIFFERENTIAL TEST, ROW 3, COLUMN 2



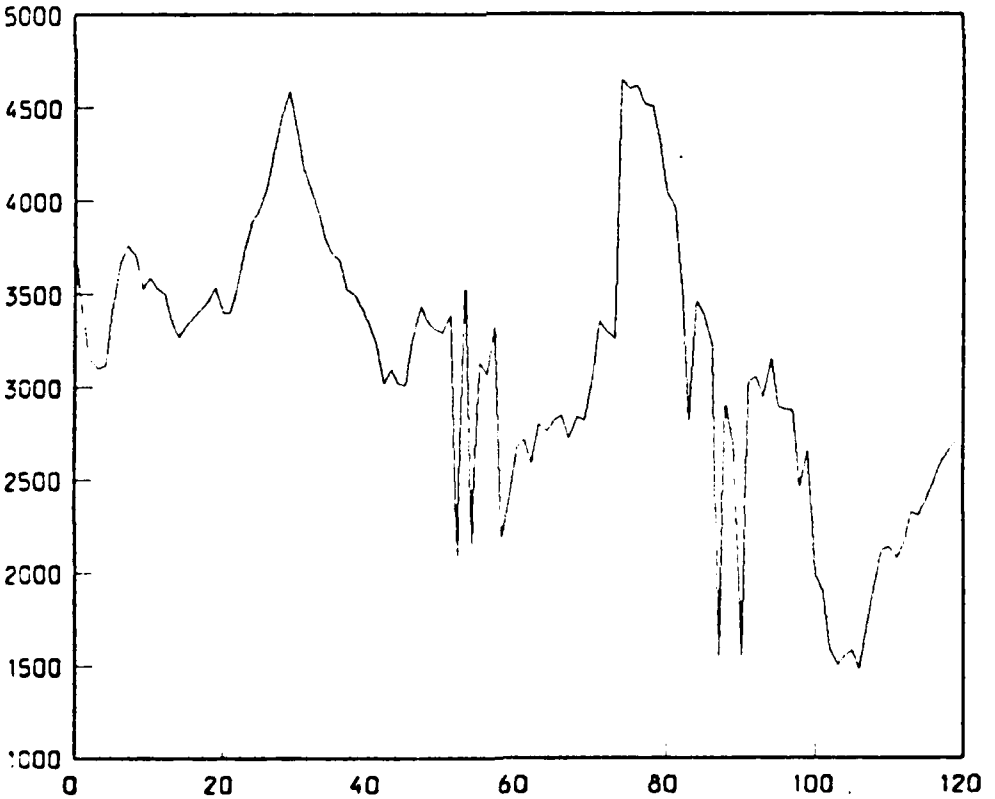
DAY 11, DIFFERENTIAL TEST, ROW 2, COLUMN 3



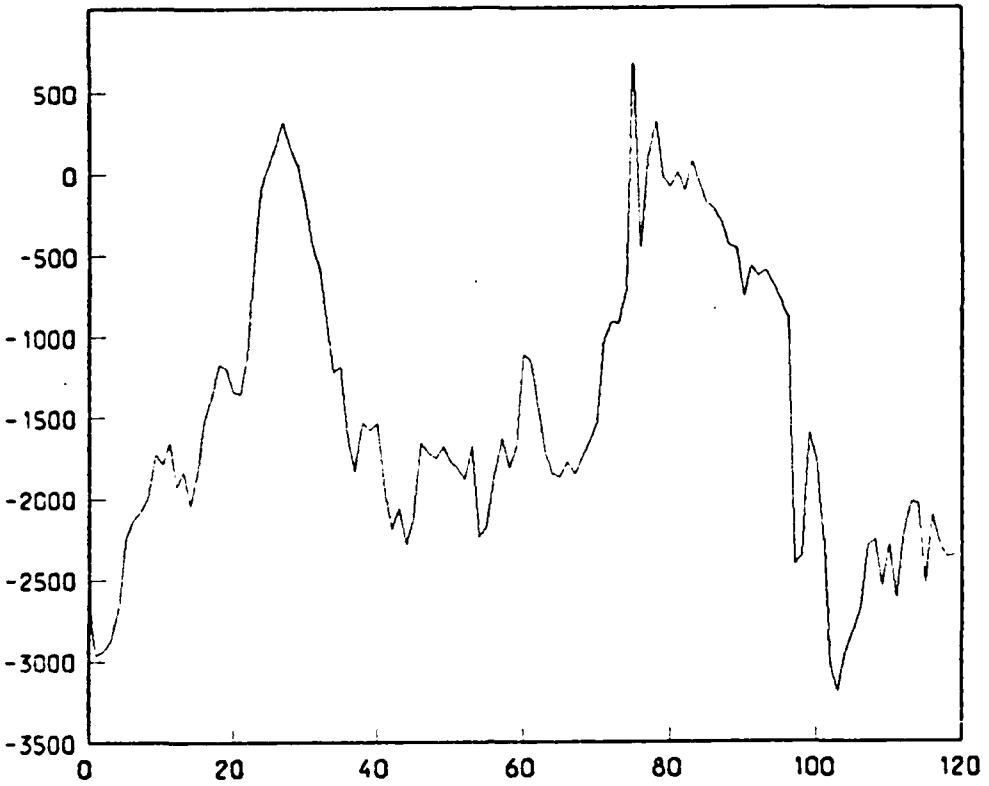
DAY 11, DIFFERENTIAL TEST, ROW 2, COLUMN 4



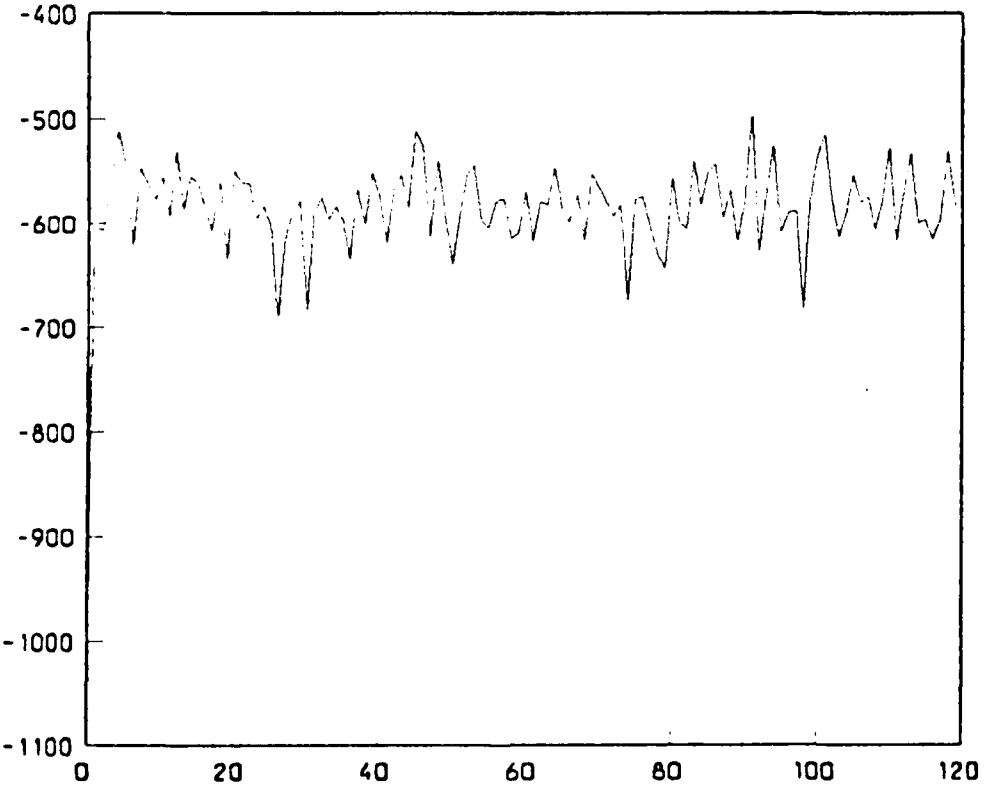
DAY 11, DIFFERENTIAL TEST, ROW 2, COLUMN 1



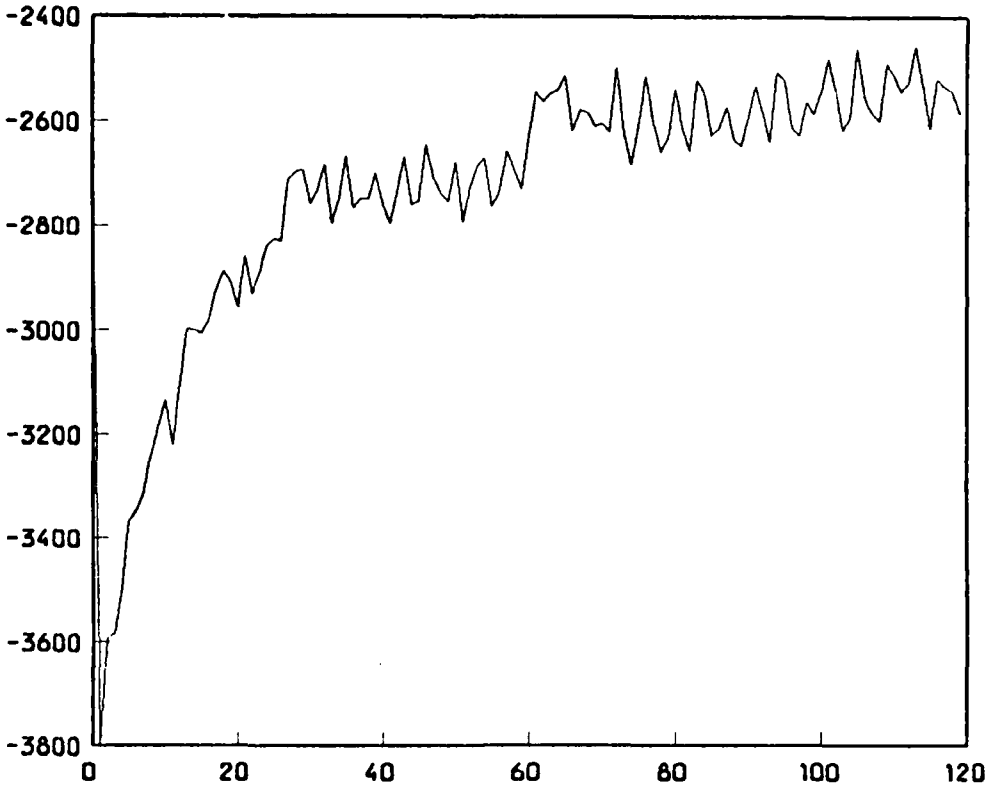
DAY 11, DIFFERENTIAL TEST, ROW 2, COLUMN 2



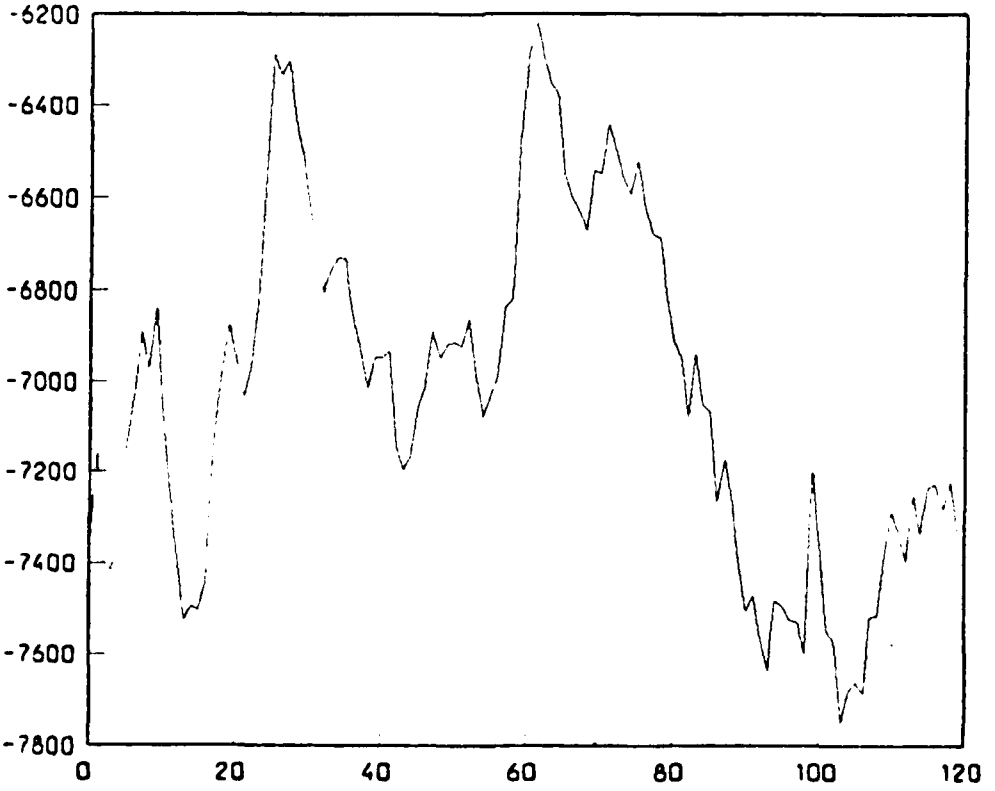
DAY 11, DIFFERENTIAL TEST, ROW 1, COLUMN 3



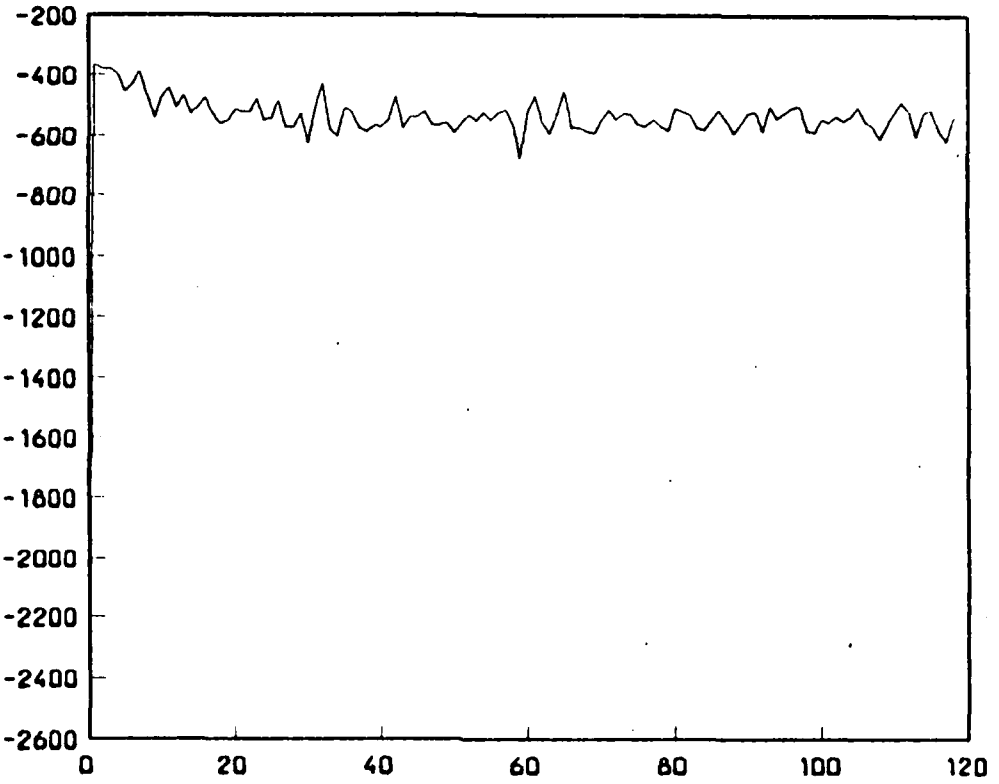
DAY 11, DIFFERENTIAL TEST, ROW 1, COLUMN 4



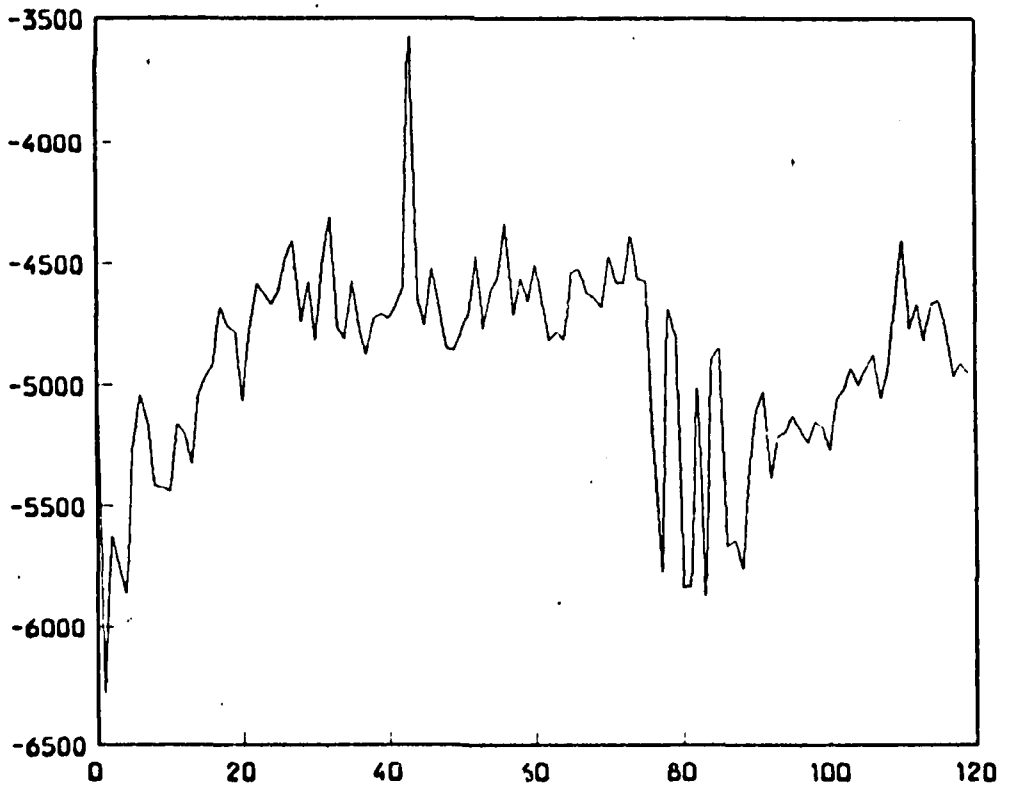
DAY 11, DIFFERENTIAL TEST, ROW 1, COLUMN 1



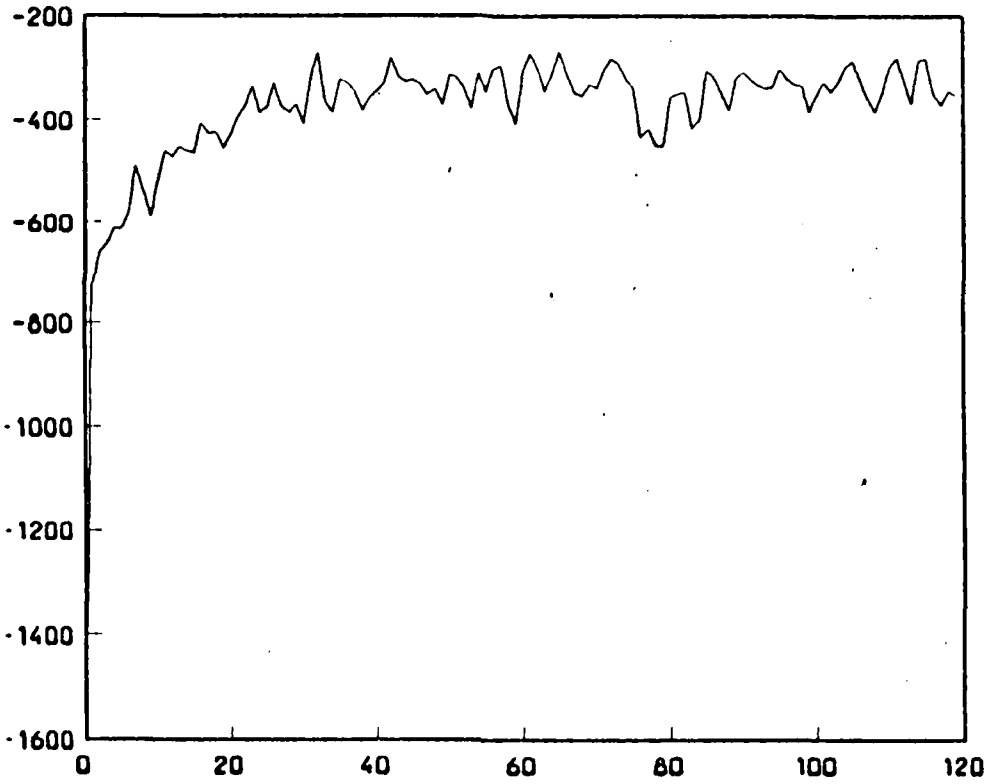
DAY 11, DIFFERENTIAL TEST, ROW 1, COLUMN 2



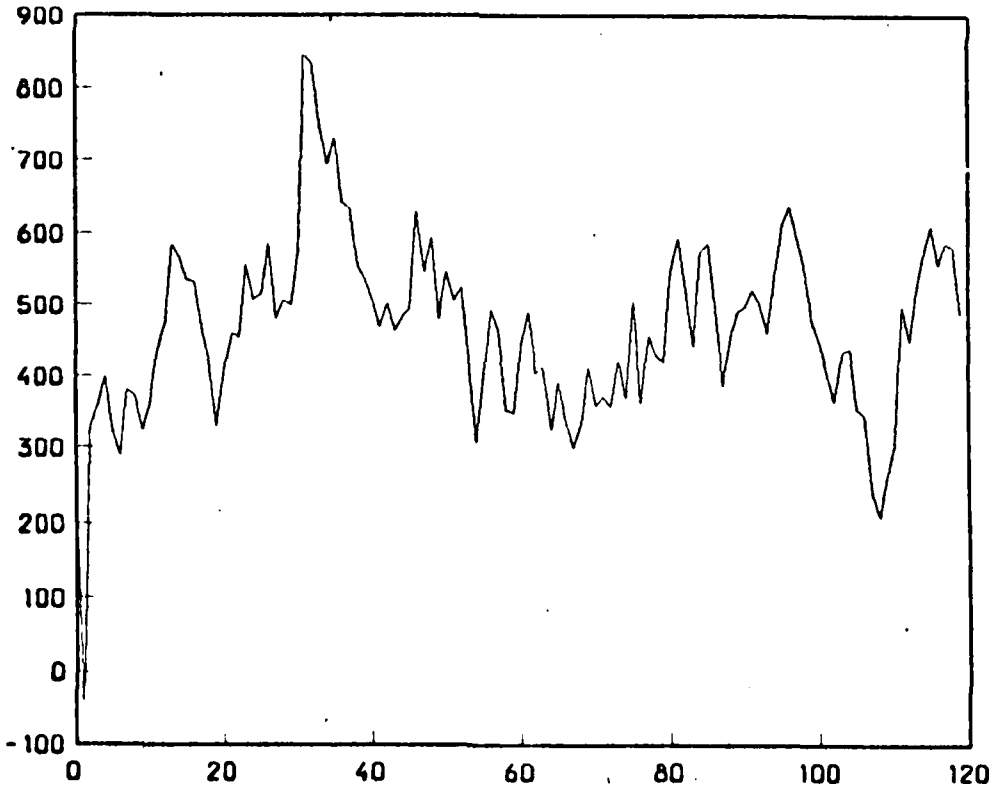
DAY 8, ABSOLUTE TEST, ROW 4, COLUMN 3



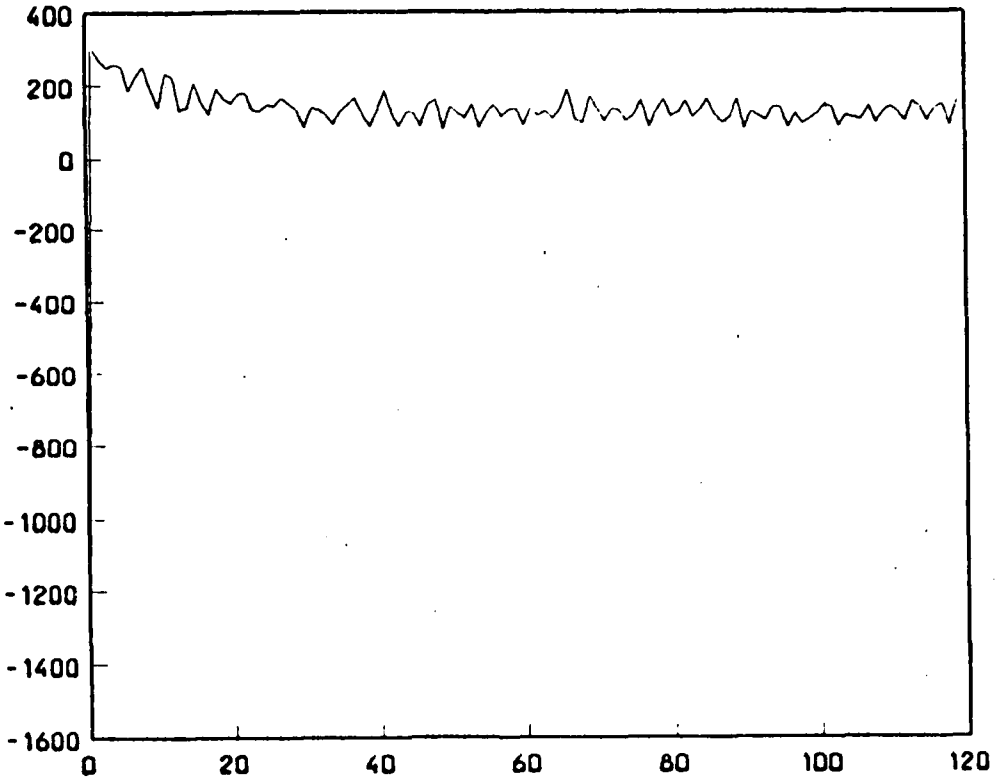
DAY 8, ABSOLUTE TEST, ROW 4, COLUMN 4



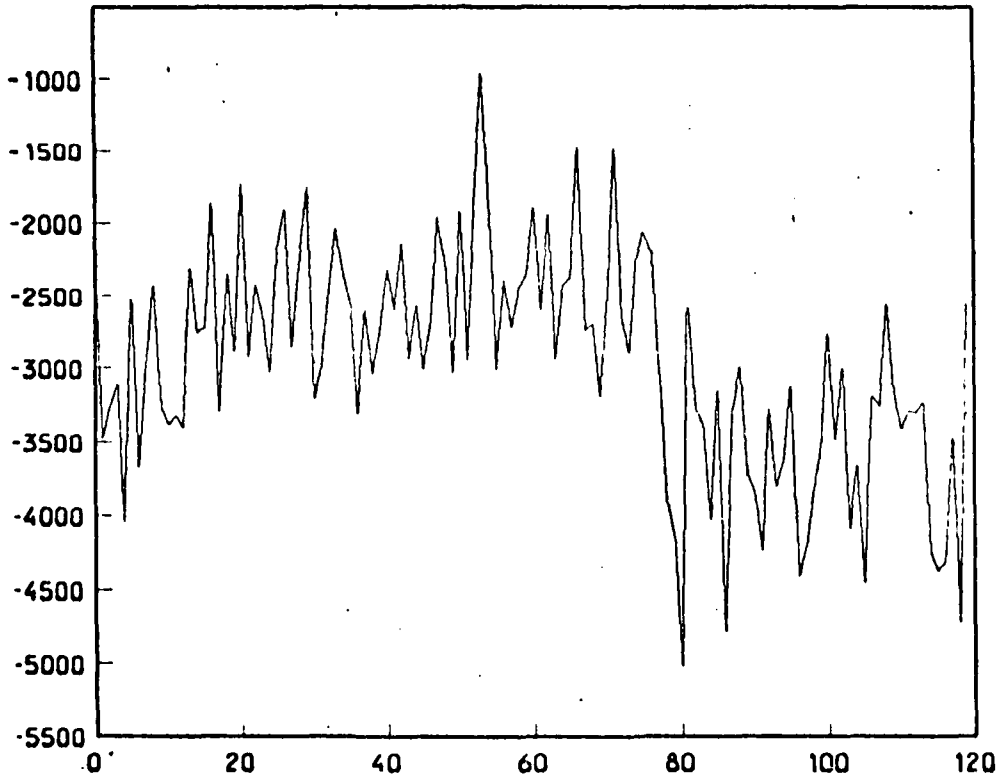
DAY 8, ABSOLUTE TEST, ROW 4, COLUMN 1



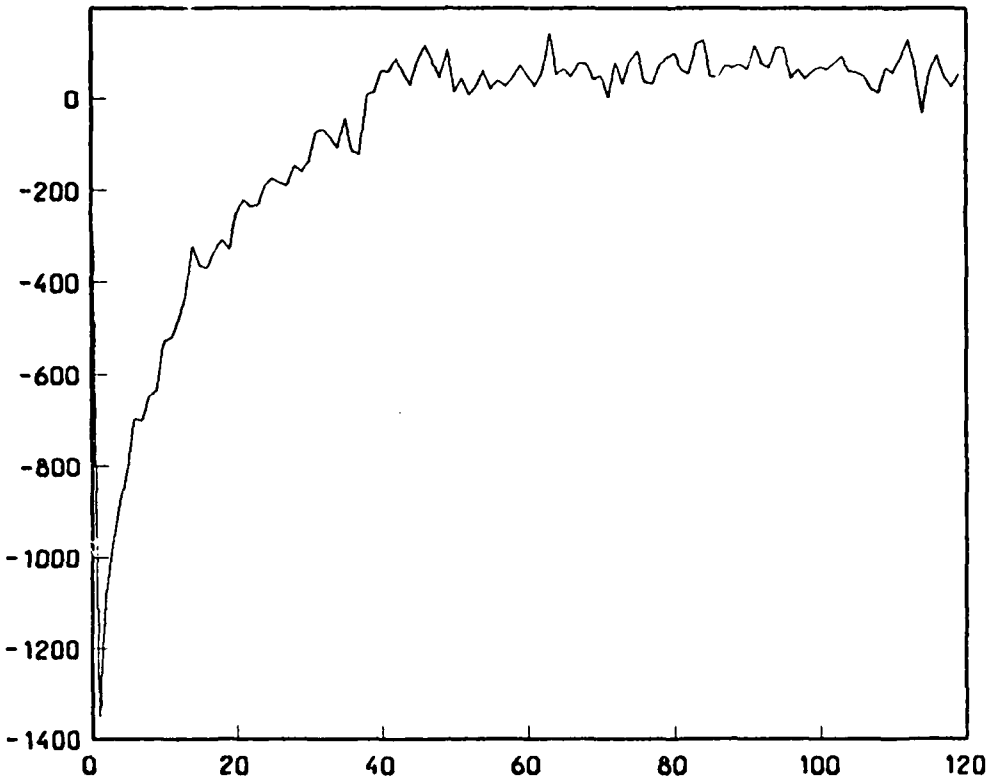
DAY 8, ABSOLUTE TEST, ROW 4, COLUMN 2



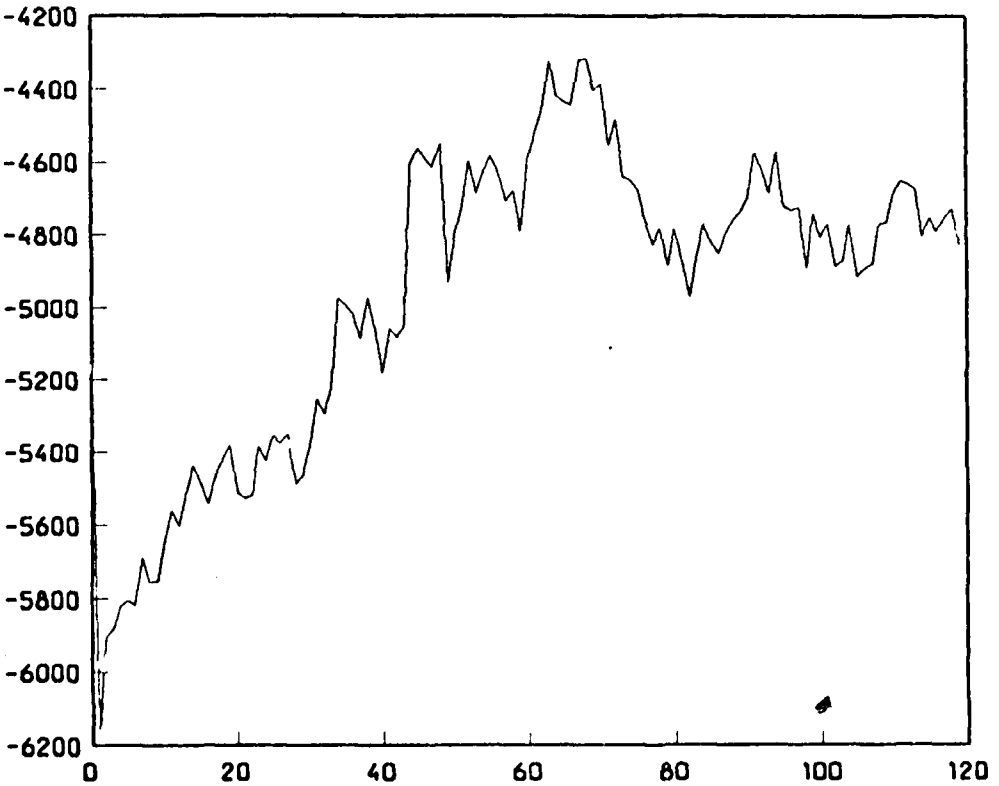
DAY 8, ABSOLUTE TEST, ROW 3, COLUMN 3



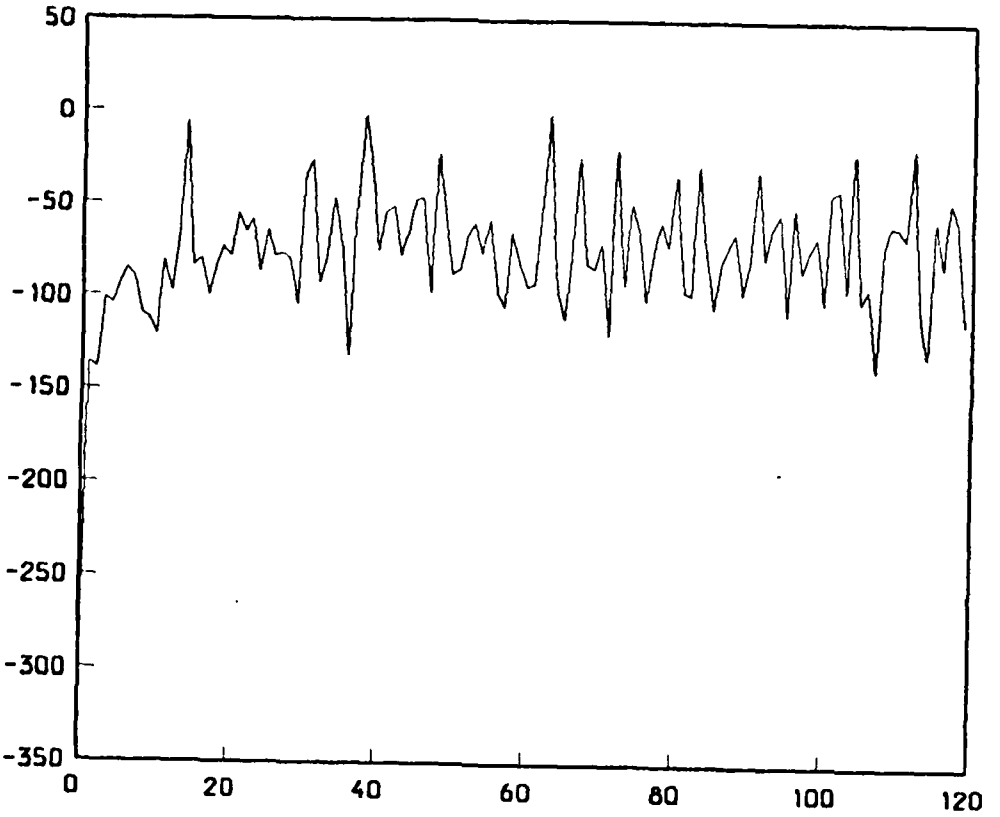
DAY 8, ABSOLUTE TEST, ROW 3, COLUMN 4



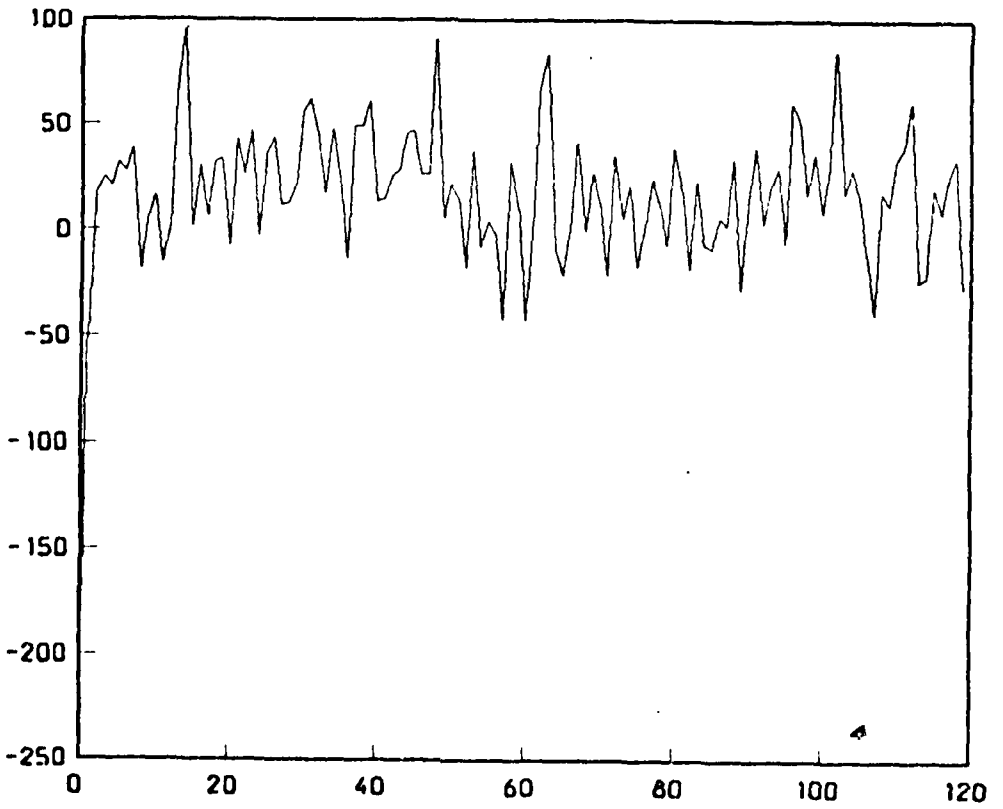
DAY 11, ABSOLUTE TEST, ROW 2, COLUMN 1



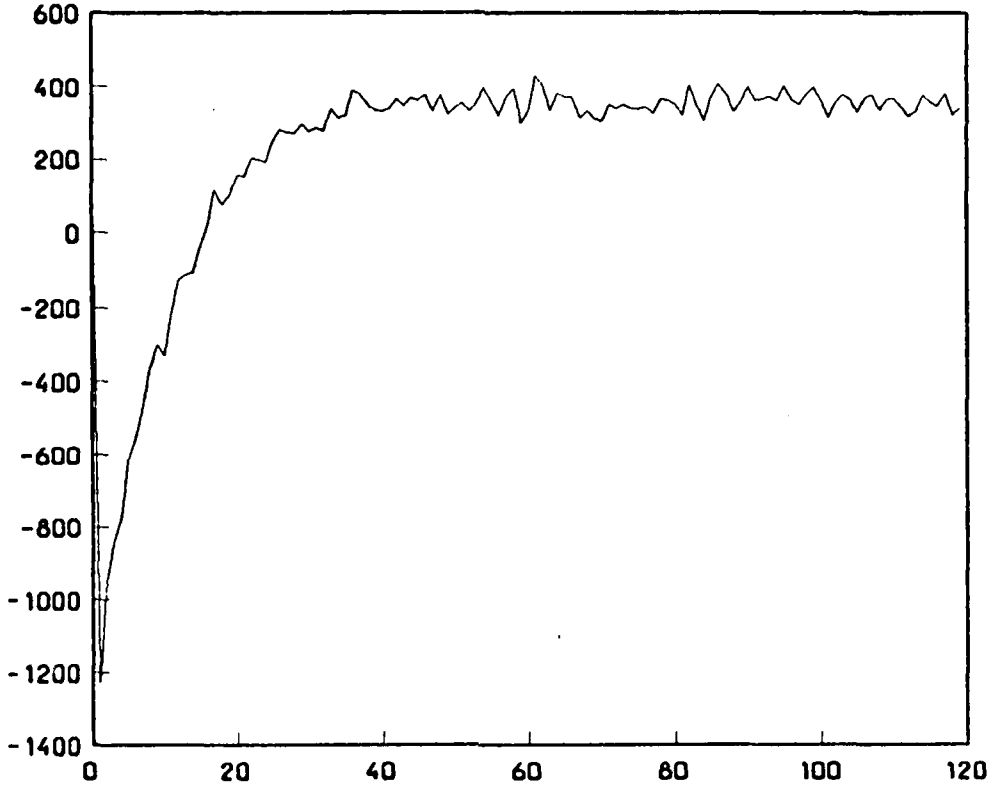
DAY 11, ABSOLUTE TEST, ROW 2, COLUMN 2



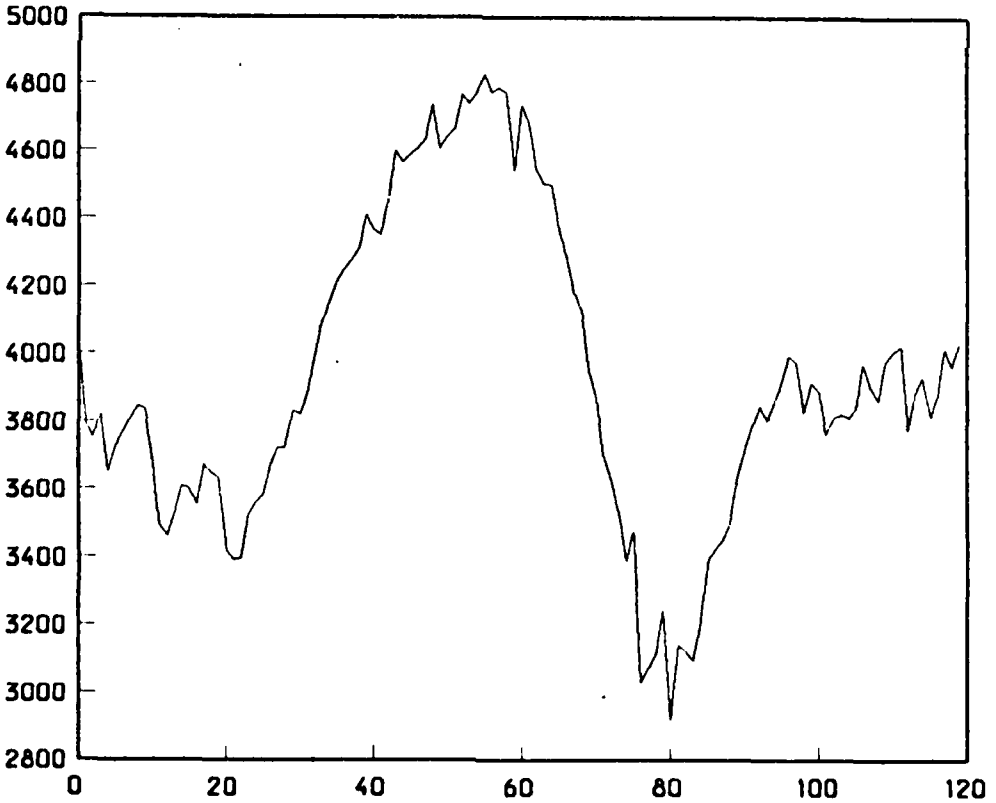
DAY 11, ABSOLUTE TEST, ROW 2, COLUMN 3



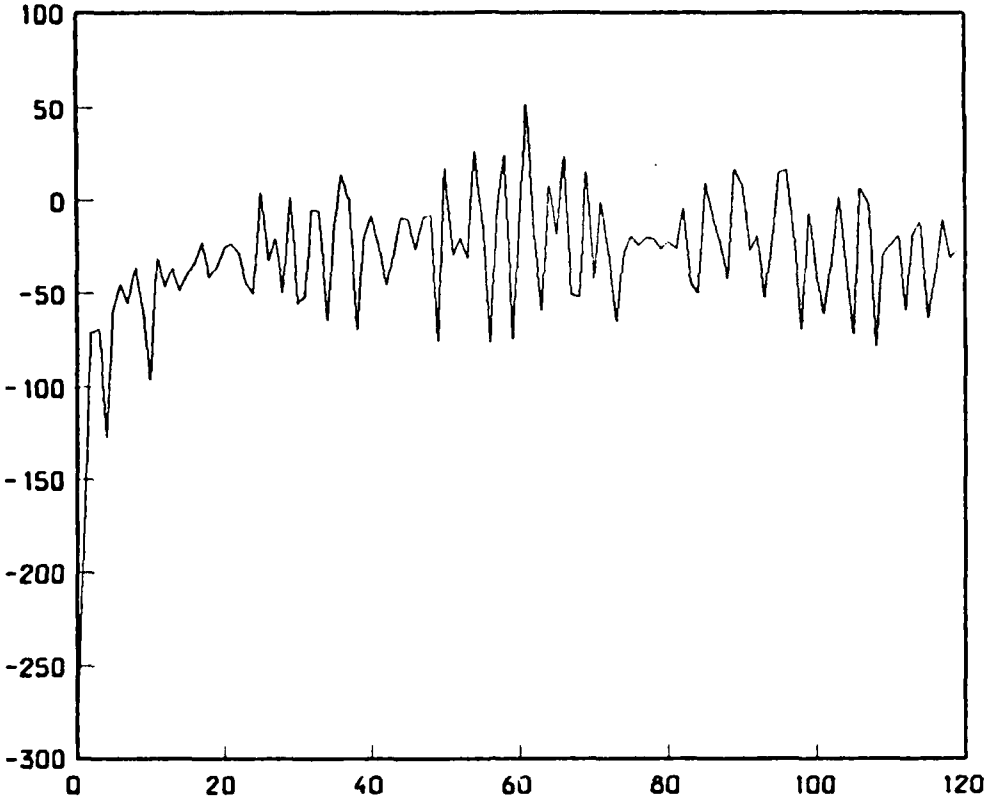
DAY 11, ABSOLUTE TEST, ROW 2, COLUMN 4



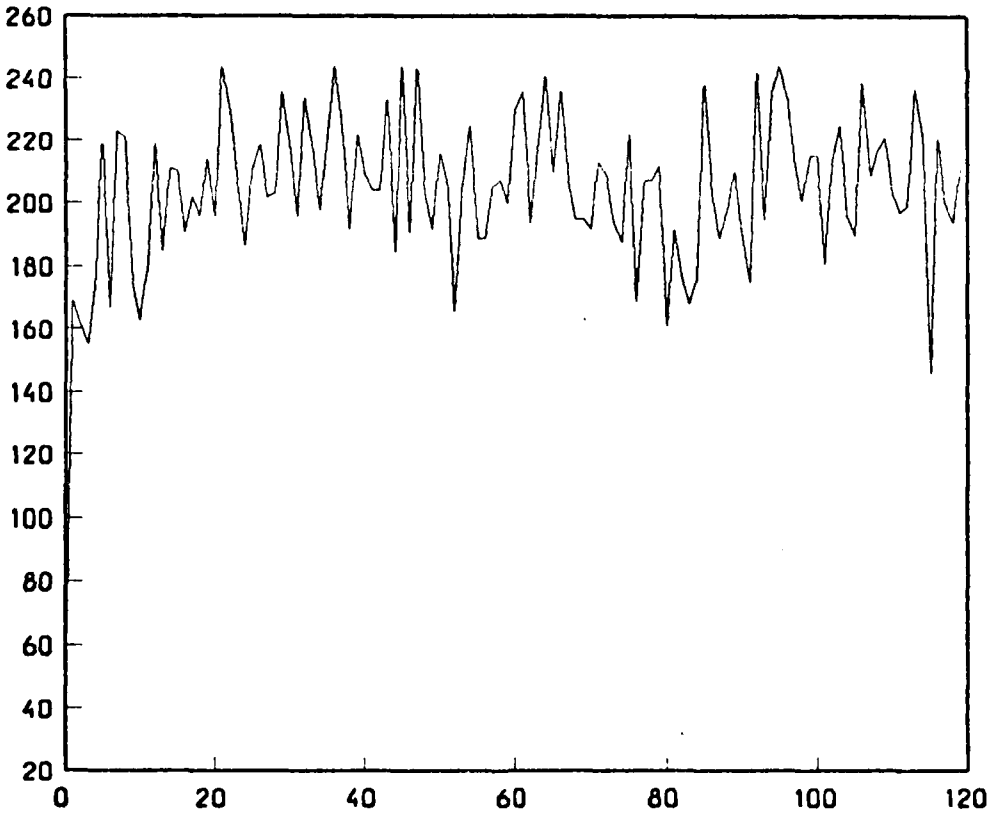
DAY 11, ABSOLUTE TEST, ROW 3, COLUMN 1



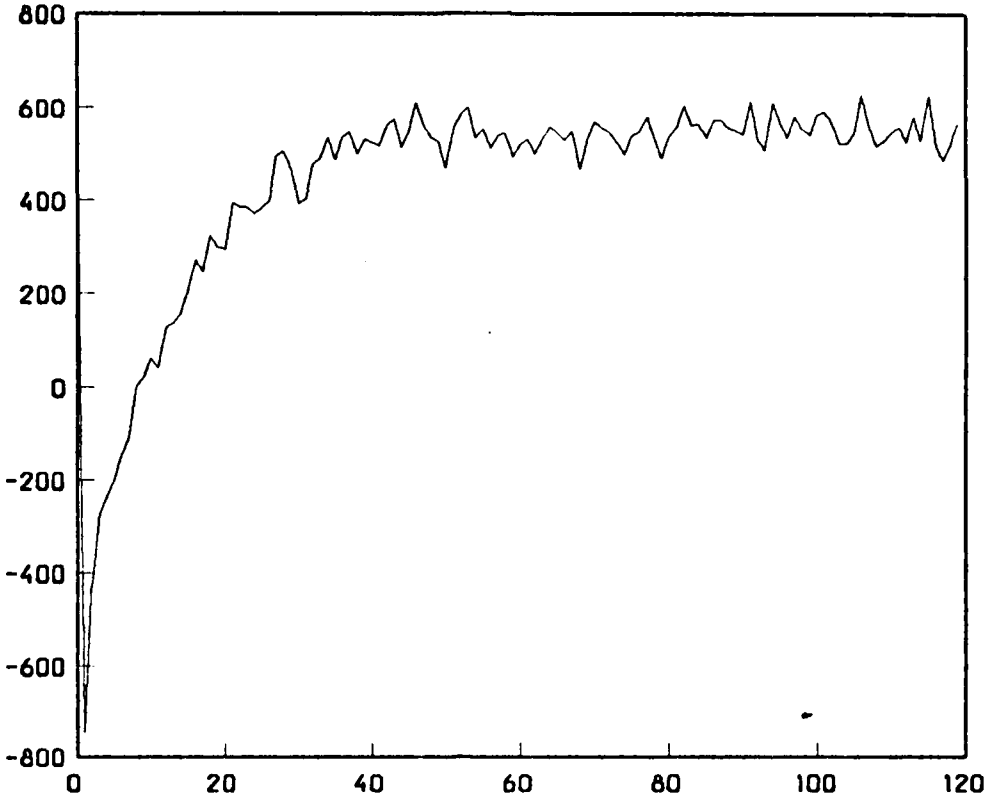
DAY 11, ABSOLUTE TEST, ROW 3, COLUMN 2



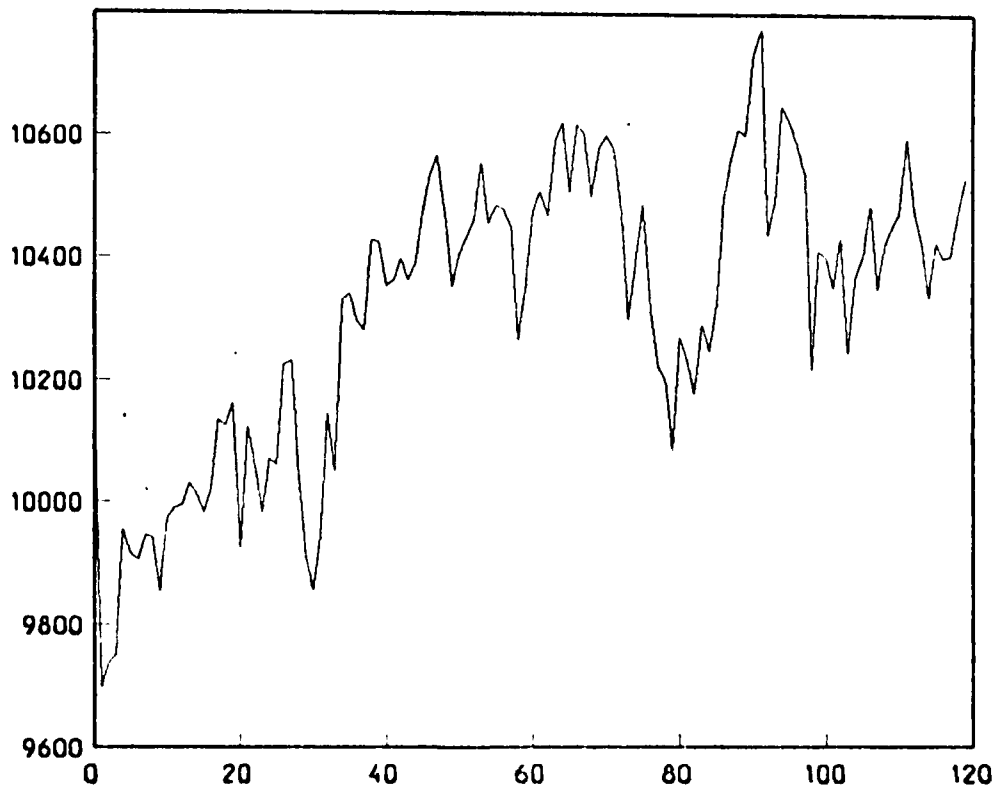
DAY 11, ABSOLUTE TEST, ROW 3, COLUMN 3



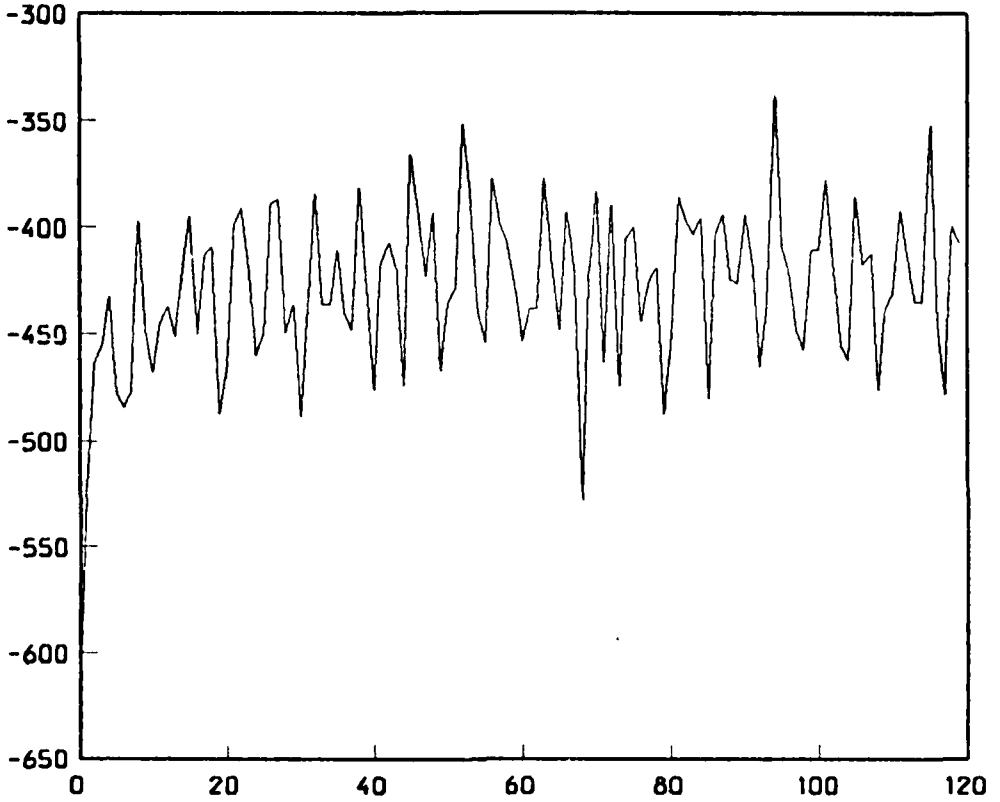
DAY 11, ABSOLUTE TEST, ROW 3, COLUMN 4



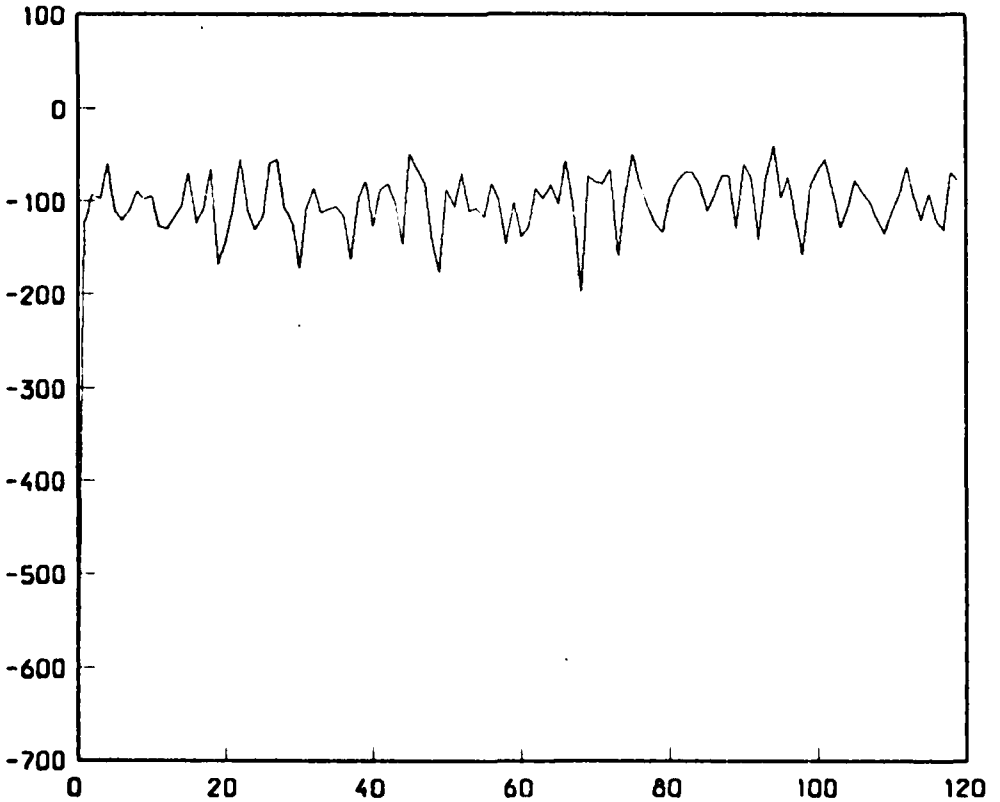
DAY 11, ABSOLUTE TEST, ROW 4, COLUMN 1



DAY 11, ABSOLUTE TEST, ROW 4, COLUMN 2



DAY 11, ABSOLUTE TEST, ROW 4, COLUMN 3



DAY 11, ABSOLUTE TEST, ROW 4, COLUMN 4

APPENDIX D

Software Programs and Commands

test.multicomm command

```
dacq -TA -D/dev/dacp/adf0  -M -cf2 -cs3 -cn5 -cv10000
  -ch250.000000 -G -n5 -r0 -il -B -d/usr/collier/filename
  -i75000 -A70.000000
```

This command line tells the computer to read in through the A/D board five channels starting with channel 0 and incrementing by one up to channel 4. The five channels are sampled at a frequency of 250 hz with a converter speed of 10000 channel/s. Two clocks are used with one triggering (or gating) the other. The computer samples 75000 bits of data. If it has not finished sampling the data in 70 seconds, it times out. The data that is sampled is stored under a specified filename.

da.command command

```
dacq -TD -D/dev/dacp/ daf0 -c16 -cH250 -cd50.000000 -n1 -f0
      -i0 -B -d/usr/collier/filename -30000 -A120.000000
```

This command tells the computer to take 30000 bits of data stored under a specified filename, at 250 hz, and run the data through the D/A converter out through channel 0.

DACQ Switches

-Af alarm timeout value in seconds

-B set main memory big buffer mode

-Ds device pathname

-F set file transfer via driver mode

-M multiple clock setup

-Tc device type (A = a/d, D = d/a)

-b differential input on A/D

-cHf clock frequency in Hz

-cdi clock duty cycle in percent

-cfi first clock number

-cli clock number to use for single channel clock

-cni number of channels for multi-clock waveform

-csi second clock number

-cvi converter speed in channels/s

-ds disk file name

-fi first channel in incremental mode

-gi gain for incremental mode

-ii increment between channels in incremental mode

-li transfer length in items

-ni number of channels in incremental mode

-rs file name for random channel list

-s simultaneous samples (D/A, A/D)

"i" in a switch listing means that the module is expecting an integer value in place of the i

"f" indicates a floating-point number

"s" indicates a string of characters, such as a disk file name

Compile command for `C' programs

```
cc filename.c -lpg -lmr -lm -o filemane.out
```

cc	Calls up the `C' compiler program.
filename.c	The name of the file written in `C'. Must end in ".c".
-lgp	Tells the compiler to call up the graphics library.
-lmr	Tells the compiler to call up the object archive library.
-lm	Tells the compiler to search the math library.
-o	Tells the compiler to output an object file.
filename.out	The name which the compiled object file is called.

Plotp Commands

-g	suppress grids
-a	suppress axes
-cs	string of plot characters.one per line; set marking
-f	suppress frame
-gxbi	bundle for x grids
-gybi	bundle for y grids
-lni,bi	bundle for plot line i
-lni,mk	mark plot line i
-ri	region to draw in
-txbi	bundle for x tics
-txsi	size for x tics (% of frame size)
-tybi	bundle for y tics
-tysi	size for y tics (% of frame size)
-xa	no x axis mark label
-xbi	bundle for x axis
-xhf	x axis high limit
-xi	x origin of plot
-xif	x axis tic/grid interval
-xlf	x axis low limit
-xni	x axis no. of tics/grids
-xt	no x axis title
-ya	no y axis mark label
-ybi	bundle for y axis
-ybf	y axis high limit
-yi	y origin of plot
-yif	y axis tic/grid interval
-ylf	y axis low limit
-yni	y axis no. of tics/grids
-yt	no y axis title
-Axbi	bundle for alternate x axis
-Aybi	bundle for alternate y axis
-Xi(P,T)s	X data file for line i, Permanent or Temporary
-Yi(P,T)s	Y data file for line i, Permanent or Temporary
-lni,mbi	text bundle for mark character for plot line i
-lni,msi	size of mark characters for plot line i

i = integer #

f = floating pt #

Array.c Program

```
/*This program is to extract from an already created binary
D/A file, a specified channel.  After extraction the channel
is displayed using graphics routines and then written to a
user given output file.  The program repeats until user wants
to stop.*/

#include <stdio.h>
#define FLBUFSIZE  600
#define BUFFSIZE   600
#define MAXCHANNELS  5
#define MINUSONE       (-1)

/*the control array for the mgiosc command*/

int control[13] = {

     0,    /* 0 starting index */
     0,    /* 1 number of 'x'points plotted, filled in later*/
     100,  /* 2 starting x value */
     0.    /* 3 starting y value */
     0x36,/* 4 master control word, enable scaling 36=11110*/
     1,    /* 5 data points per increment */
     1,    /* 6 increment value */
     0,    /* 7 offset, filled in later by scaling routine */
     1,    /* 8 scaling multiplier */
     1,   /* 9 scaling divisior filled in later by scaling
             routine */
     60,   /* 10 secondary addition, used to raise off bottom
              of screen */
     0,    /* 11 sync windo width */
     0,    /* 12 sync value */
};

int  xl,yt,yr,yb,null;

main ()

{

     int   fb,channum,DONE;
     char  data_in_file[30],data_out_file[30];
     short  int  gpbuff[BUFFSIZE];
     int  nchan;
     static short  filebuf[FLBUFSIZE * MAXCHANNELS];
     int c;
     int  ENDFILE = 0;
     int  bufferflag;
     char  stopchar;
```

```
      int  j;
      int   count,count2;
      int  vunum = 2;
      int fpin, fpout;  /*for use with read and write */
      int  n_bytes, data_items;

/* initialize graphics, files, how many channels per how many
frames*/

DONE = 0;  /* flag to let user quit */

/* get info from user */

printf("Before continuing I need some information from you.\n\n");
printf("This program reads from an EXISTING binary file, \n");
printf("displays a selected channel, then writes that channel\n");
printf("information to a given output file.   ");

     mgiasngp(0,0);

while(DONE ==0)

   {
     /* this user info inside loop, will need for every
     extraction */

     printf(" I need from you the following;  \n\n");
     do
     {
     printf("   The number of channels per frame;  ");
     scanf("%d",&nchan);
     if(nchan <= 0 || nchan > MAXCHANNELS)
printf("\nNumber of channels is between 0 and %d \n",MAXCHANNELS);
  }
     while(nchan <=0 || nchan > MAXCHANNELS);
     do
     {
printf("   The channel number to extract   ");
     scanf("%d",&channum);
     }
     while(channum <= 0 || channum > nchan);
     channum = channum - 1;/* bring from 1 - 4 to 0 - 3range */
printf("    The EXISTING binary input file name:   ");
  scanf("%s",data_in_file);
     fpin = open(data_in_file,0);  /*0 - reads */

     if (fpin == MINUSONE)
       {
          printf("Error in opening %s  , \n",data_in_file);
          exit(1);
       }

printf("    The output file name:  ";
```

```
      scanf("%a",data_out_file);

      if((fpout = open(data_out_file,1)) == -1)
        fpout = creat(data_out_file,0766);

      if(fpout == MINUSONE)
        {
printf("Error in opening %s  . \n",data_out_file);
          exit(1);
        }

/*init graphics  */

      mgipin(-1);  /* enable all planes  */
      mgihue(3);
      mgifb(1,3);
      mgiclearpin(2,-1,0);
      fb = 1;
      mgifb(3-fb,fb);
      mgigetvcoor(vunum,&xl,&yb,&xr,&yt,&null); /*get windo
                                        coordinates for scaling*/

 while((n_bytes = read(fpin, filebuf,(FLSBUFSIZE*nchan*2))) > 0)

/* do the extraction, graphics and writing */

      {
        count2 = n_bytes/2;

/*extract the channel from the file buffer  */

      if(count2 > 600)
           count2 = count2/nchan; /*do only 600 max for graphics
                                  or do up to what ever count2 is*/

      for(count = 0;count < count2; count++)
      {
          gpbuff[count] = filebuf[(count*nchan)+channum];
/*printf("gpbuff[ %d ] =   %d  \n",cont,gpbuff[count]);*/
      }

/* count now contains the number of items in the graphics
buffer count is <= 600 will scale only the number of points
obtained.  except for last of file this number should be 600.
*/

/* now scale the data to the windo  */

      scale_data(gpbuff,count);
      control[1] = count;
      bufferflag = 1;
      mgiosc(gpbuff,control);
      mgifb(fb,3-fb);
```

```
      mgiclearpln(2,-1,0);
      mgaaol(18,mgfadd(&bufferflag));

      while(bufferflag != 0)  ;
        fb = 3 - fb;
/*write gpbuff to the data_out_file */

/* for(j=0;j<count;j++)
         putc(gpbuff[j],fpout);  */
      write(fpout,gpbuff,(BUFFSIZE * 2));

      }  /* end while */

/* close the files */

      close(fpin);
      close(fpout);
      endfile = 0; /* RESET THE END OF THE FILE MARKER */
      mgiclearpln(2,-1,0);

printf("Would you like to do another chan ??? 1=Y or 2=N \n\n;
      scanf("%d",&j);
        if(j ==2) /* || (j=="N"))  */
           DONE = 1;
        else
           DONE = 0; /* dont stop */

}  /* end while */

      mgifb(1,3);
      mgiclearpln(2,-1,0);
      mgideagp();

}  /*end main */

/*****************************************************************/

scale_data(gpbuff,count)

      short  *gpbuff;
      int  count:

{
      int  maximum, minimum, old_maximum;

        maximum = findmax(gpbuff,count);
        minimum = findmin(gpbuff,count);
        old_maximum = maximum;

/*make data zero relative */

      if(minimum < 0)
        control[7] = abs(minimum);
```

33	daylldfr1c1 daylldfr1c3	daylldfr1c2 daylldfr1c4	strobelldfr1
34	daylldfr2c1 daylldfr2c3	daylldfr2c2 daylldfr2c4	strobelldfr2
35	daylldfr3c1 daylldfr3c3	daylldfr3c2 daylldfr3c4	strobelldfr3
36	daylldfr4c1 daylldfr4c3	daylldfr4c2 daylldfr4c4	strobelldfr4
37	dayllr5	strobedllr5	

15	day6r4c1 day6r4c3	day6r4c2 day6r4c4	strobed6r4
16	day7dfr2c1 day7dfr2c3	day7dfr2c2 day7dfr2c4	strobed7dfr2
17	day7dfr3c1 day7dfr3c3	day7dfr3c2 day7dfr3c4	strobed7dfr3
18	day7dfr4c1 day7dfr4c3	day7dfr4c2 day7dfr4c4	strobed7dfr4
19	day7bucr4c1 day7bucr4c3	day7bucr4c2 day7bucr4c4	strobebcd7r4
20	day7abr4c1 day7abr4c3	day7abr4c2 day7abr4c4	strobed7abr4
21	day8dfr1c1 day8dfr1c3	day8dfr1c2 day8dfr1c4	strobed8dfr1
22	day8dfr2c1 day8dfr2c3	day8dfr2c2 day8dfr2c4	strobed8dfr2
23	day8dfr3c1 day8dfr3c3	day8dfr3c2 day8dfr3c4	strobed8dfr3
24	day8dfr4c1 day8dfr4c3	day8dfr4c2 day8dfr4c4	strobed8dfr4
25	day8abr1c1 day8abr1c3	day8abr1c2 day8abr1c4	strobed8abr1
26	day8abr2c1 day8abr2c3	day8abr2c2 day8abr3c4	strobed8abr2
27	day8abr3c1 day8abr3c3	day8abr3c2 day8abr3c4	strobed8abr3
28	day8abr4c1 day8abr4c3	day8abr4c2 day8abr4c4	strobed8abr4
29	day11abr1c1 day11abr1c3	day11abr1c2 day11abr1c4	strobe11abr1
30	day11abr2c1 day11abr2c3	day11abr2c2 day11abr2c4	strobe11abr2
31	day11abr3c1 day11abr3c3	day11abr3c2 day11abr3c4	strobe11abr3
32	day11abr4c1 day11abr4c3	day11abr4c2 day11abr4c4	strobe11abr4

APPENDIX E

List Of Files On Floppy Disks

Disk #	Files On Disk		
1	day3r1c1	day3r1c2	strobed3r1
	day3r1c3	day3r1c4	
	day3r2c1	day3r2c2	strobed3r2
	day3r2c3	day3r2c4	
2	dayer3c1	day3r3c2	strobed3r3
	day3r3c3	day3r3c4	
3	day3r4c1	day3r4c2	strobed3r4
	day3r4c3	day3r4c4	
4	day4dfr1c1	day4dfr1c2	strobed4dfr1
	day4dfr1c3	day4dfr1c4	
5	day4dfr2c1	day4dfr2c2	strobed4dfr2
	day4dfr2c3	day4dfr2c4	
6	day4dfr3c1	day4dfr3c2	strobed4dfr3
	day4dfr3c3	day4dfr3c4	
7	day4dfr4c1	day4dfr4c2	strobed4dfr4
	day4dfr4c3	day4dfr4c4	
8	day5r1c1	day5r1c2	strobed5r1
	day5r1c3	day5r1c4	
9	day5r2c1	day5r2c2	strobed5r2
	day5r2c3	day5r2c4	
10	day5r3c1	day5r3c2	strobed5r3
	day5r3c3	day5r3c4	
11	day5r4c1	day5r4c2	strobed5r4
	day5r4c3	day5r4c4	
12	day6r1c1	day6r1c2	strobed6r1
	day6r1c3	day6r1c4	
13	day6r2c1	day6r2c2	strobed6r2
	day6r2c3	day6r2c4	
14	day6r3c1	day6r3c2	strobed6r3
	day6r3c3	day6r3c4	

```
/*This is the subroutine which is called to find the first
strobe pulse. It then returns to the main program with the
pointer set.*/

int  findstrb(strobuff)
short  strobuff[];
{
   int  i,j;
   short  *bp1, *bp2;
     bp1 = strbbuff;
     for(j=0;j<STRBBUFFSIZE;j++)
      {
         if(*bp1 > -100 && *bp1 < 100)
           {
              return(j);
           }
         bp1++;
      } /*end for loop*/
   return(-1);
}/*end findstrb*/
```

```
recognized.*/

if(*strbptrl > -100 && *strbptrl < 100 && nottop ==1 &&
          optndx > 100)
     {
          optndx = 0;
          nottop = 0;
          strobecnt++;
     }
if(*strbptrl < -100)
          nottop = 1;

if((ndata_bytes !=  && nstrb_bytes != 0))
     {
       A[optndx] = A[optndx] + databuf[datandx];
       optndx++;
       datandx++;
       stbndx++;
       strbptrl++;
     }

/* end while */
   for(i = 0; i < BUFFSIZE; i++)
   A[i] = A[i]/strobecnt;

/*This section writes the summed array out to the output file
and closes out the pointers. It then calls up the plot program
and querries whether the operator wants to sum another file.*/

write(fpout,A,(120*4));
close(fpdata);
close(fpstrb);
close(fpout);

printf("Plotting the output file \n");

strcat(BTOA,data_out_file);
strcat(BTOA,"   >tempfile\n");
system(BTOA);
system(systemtext);
strcpy(data_out_file,blanks);
strcpy(BTOA,"btoa");
for(i=0;i<BUFFSIZE;i++)
   A[i] = 0;
   strobecnt = 0;
printf("Wouldyoulike tosumanotherfile??? 1=Y or 2=N\n\n");
  scanf("%d",&j);
   if(j == 2)
      DONE = 1;
   else
      DONE = 0;
  } /*end while*/
} /*end main*/
```

```
            exit(1);
          }
      }
    else
        {
          printf("Strob file is empty\n");
          exit(1);
        }

nottop = 0; /*must be at the top of a strobe or findstrb would
              have returned a -1 and then exit before here*/

/*This section reads in a buffer of the data file and set the
datandx to the stbndx so that the data corresponds to the
first strobe pulse.*/

ndata_bytes = read(fpdata,databuf,(FLBUFSIZE * 2));
datandx = stbndx;
strbptrl = &(strbbuff[stbndx]);
printf("\n\n Starting summations. Please wait.\n");

/* Here begins a while loop which continues as long as there is
data in the strobe file and the data file to read, or until
strobecnt is equal to the number of iterations specified.*/

while (ndata_bytes > 0 && nstrb_bytes > 0 && strobecnt < iters)
   {
/*test to see if need to read more data*/
     if(datandx >= (ndata_bytes/2))
       {
         ndata_bytes = read(fpdata,databuf,(FLBUFSIZE * 2));
         datandx = 0;
       }

/*test to see if need to read more strobe file */
     if(stbndx >= (nstrb_bytes/2))
        {
          nstrb_bytes= read(fpstrb,strbbuff,(STRBBUFFSIZE*2));
          strbptrl = strbbuff;
          stbndx = 0;
        }

/*This section checks to find the next strobe pulse and sums
the values in the array.  If a strobe pulse is not found,
the counters are increased by 1."optndx" must have advanced a
minimum of 100 data points before it will recognize anotner
strobe pulse.  This tries to guarantee that no unusually high
noise is recognized as a strobe pulse, but it allows a
variation in strobe period. Once another strobe is recognized
"optndx" is set to 0 and "nottop" is set to 0 signifying the
strobe pulse top again. Once the strobe pulse falls again,
"nottop" is set to 1, signifying a low stat of the strobe.
This is necessary to occur before another strobe pulse is
```

```
        exit (1);
      }
printf("\nThe output file name:    ");
  scanf("%s",data_out_file);
  if((fpout = open(data_out_file,1)) == -1)
    fpout = creat(data_out_file,0766);
    if (fpout == MINUSONE)
       {
          printf("Error in opening %s   .\n",data_out_file);
          exit (1);
       }
printf("\nThe strobe file name:    ");
  scanf("%s",strob_file_in);
  if((fpstrb = open(strb_file_in,0)) == -1)
     {
       printf("Error in opening %s   .\n",strb_file_in);
       exit(1);
     }
printf("\nHow many iterations do you want?   ");
  scanf("%d",&iters);
printf("\nIs the plot to be printed? 1 Yes or 2 No   ");
  scanf("%s",plotorprint);
  if(plotorprint[0] == '1')
    {
      strcpy(systemtext,PRINTTEXT);
    }
  else
    {
      strcpy(systemtext,SCREENTEXT);
    }

/*This section initializes the variables which need to be*/

stbndx = 0;
datandx = 0;
optndx = 0;
A[BUFFSIZE] = 0;
strobecnt = 0;

/*This section reads in a buffer of the strobe file and calls
a subroutine 'findstrb' to locate the first strobe. If no
strobe values are found or if the file is empty, the
appropriate response is given and the program is exited. If a
strobe is found, 'nottop' is set to 0 to indicate that we are
at the top of the strobe. 'findstrb' also returns with the
stbndx set to a number so we know when the first strobe
begins.*/

if((nstrb_bytes = read(fpstrb,strbbuff,(STRBBUFFSIZE*2))) > 0)
   {
      if((stbndx = findstrb(strbbuff)) == -1)
        {
           printf("No strobe in strobe file\n");
```

Varisum.c Program

```
/*This program does the same thing as the strobesum.c program
above, except it allows the operator to specify how many
summations he/she desires it to perform, and plots the output
as usual*/

/*This section defines or delares all variables*/
#include <stdio.h>
#define FLBUFSIZE   125
#define BUFFSIZE     125
#define STRBBUFFSIZE FLBUFSIZE*2
#define MINUSONE    (-1)
#define PRINTTEXT  "plotp -YlPtempfile -xh100  -g lmxd\n"
#define SCREENTEXT "plotp -YlPtempfile -xl00 -g lmmd\n"

char  systemtext[50];
char  BTOA[50] = {"btoa "};
char  plotorprint[2];
char  blanks[30] = {" "};
int  i,j,DONE,strndx,datandx,optndx;
int  nstrb_bytes,ndata_bytes;
int  A[BUFFSIZE] = 0;
short  strbbuff[STRBBUFFSIZE];
static short  databuf[FLBUFSIZE];
int  fpstrb,fpdata,fpout;
char  data_in_file[30],data_out_file[30];
char  strb_file_in[30];
short  *strbptr1, strbptr2;
int  nottop;  /*if a 1 at neg of strobe, 0 means at top of
                                                  strobe*/
int  strobecnt
int  iters;

main()
{
  while (DONE ==0)
    {
/*This section queries the operator as to which files should
be opened or created, and then opens or creates them.Error
messages are given if an incorrect response is given and then
the program exits.This section also queries the operator if
the output plot is to be sent to the printer.*/

printf("\nPlease name Existing binary input file name:   ");
  scanf("%s",data_in_file);
  fpdata = open(data_in_file,0);
  if (fpdata == MINUSONE)
     {
        printf("\nError in opening %s   .\n",data_in_file);
```

```
  scanf("%d",&j);
   if(j == 2)
      DONE = 1;
   else
      DONE = 0;
  } /*end while*/
} /*end main*/


/*This is the subroutine which is called to find the first
strobe pulse. It tnen returns to the main program with the
pointer set.*/

int  findstrb(strobuff)
short  strobuff[];
{
   int  i,j;
   short  *bp1, *bp2;
     bp1 = strbbuff;
     for(j=0;j<STRBBUFFSIZE;j++)
      {
         if(*bp1 > -100 && *bp1 < 100)
           {
              return(j);
           }
         bp1++;
      } /*end for loop*/
   return(-1);
}/*end findstrb*/
```

```
strobe pulse.  This tries to guarantee that no unusually high
noise is recognized as a strobe pulse, but it allows a
variation in strobe period. Once another strobe is recognized
"optndx" is set to 0 and "nottop" is set to 0 signifying the
strobe pulse top again. Once the strobe pulse falls again,
"nottop" is set to 1, signifying a low stat of the strobe.
This is necessary to occur before another strobe pulse is
recognized.*/

if(*strbptrl > -100 && *strbptrl < 100 && nottop ==1 &&
          optndx > 100)
    {
          optndx = 0;
          nottop = 0;
          strobecnt++;
    }
if(*strbptrl < -100)
          nottop = 1;

if((ndata_bytes !=  && nstrb_bytes != 0))
    {
      A[optndx] = A[optndx] + databuf[datandx];
      optndx++;
      datandx++;
      stbndx++;
      strbptrl++;
    }

/* end while */
   for(i = 0; i < BUFFSIZE; i++)
   A[i] = A[i]/strobecnt;

/*This section writes the summed array out to the output file
and closes out the pointers. It then calls up the plot program
and querries whether the operator wants to sum another file.*/

write(fpout,A,(120*4));
close(fpdata);
close(fpstrb);
close(fpout);

printf("Plotting the output file \n");

strcat(BTOA,data_out_file);
strcat(BTOA,"   >tempfile\n");
system(BTOA);
system(systemtext);
strcpy(data_out_file,blanks);
strcpy(BTOA,"btoa");
for(i=0;i<BUFFSIZE;i++)
   A[i] = 0;
   strobecnt = 0;
printf("Wouldyoulike tosumanotherfile??? 1=Y or 2=N\n\n");
```

```
stbndx set to a number so we know when the first strobe
begins.*/

if((nstrb_bytes = read(fpstrb,strbbuff,(STRBBUFFSIZE*2))) > 0)
   {
      if((stbndx = findstrb(strbbuff)) == -1)
        {
           printf("No strobe in strobe file\n");
           exit(1);
        }
   }
  else
       {
         printf("Strob file is empty\n");
         exit(1);
       }

nottop = 0; /*must be at the top of a strobe or findstrb would
              have returned a -1 and then exit before here*/

/*This section reads in a buffer of the data file and set the
datandx to the stbndx so that the data corresponds to the
first strobe pulse.*/

ndata_bytes = read(fpdata,databuf,(FLBUFSIZE * 2));
datandx = stbndx;
strbptr1 = &(strbbuff[stbndx]);
printf("\n\n Starting summations. Please wait.\n");

/*Here begins a while loop which continues as long as there is
data in the strobe file and the data file to read.*/

while (ndata_bytes > 0 && nstrb_bytes > 0)
   {
/*test to see if need to read more data*/
     if(datandx >= (ndata_bytes/2))
       {
         ndata_bytes = read(fpdata,databuf,(FLBUFSIZE * 2));
         datandx = 0;
       }

/*test to see if need to read more strobe file */
     if(stbndx >= (nstrb_bytes/2))
        {
          nstrb_bytes= read(fpstrb,strbbuff,(STRBBUFFSIZE*2));
          strbptr1 = strbbuff;
          stbndx = 0;
        }

/*This section checks to find the next strobe pulse and sums
the values in the array.  If a strobe pulse is not found,
the counters are increased by 1."optndx" must have advanced a
minimum of 100 data points before it will recognize another
```

```
messages are given if an incorrect response is given and then
the program exits.This section also queries the operator if
the output plot is to be sent to the printer.*/

printf("\nPlease name Existing binary input file name:   ");
  scanf("%s",data_in_file);
  fpdata = open(data_in_file,0);
  if (fpdata == MINUSONE)`
     {
        printf("\nError in opening %s   .\n",data_in_file);
        exit (1);
     }
printf("\nThe output file name:   ");
  scanf("%s",data_out_file);
  if((fpout = open(data_out_file,1)) == -1)
    fpout = creat(data_out_file,0766);
    if (fpout == MINUSONE)
       {
          printf("Error in opening %s   .\n",data_out_file);
          exit (1);
       }
printf("\nThe strobe file name:   ");
  scanf("%s",strob_file_in);
  if((fpstrb = open(strb_file_in,0)) == -1)
     {
       printf("Error in opening %s   .\n",strb_file_in);
       exit(1);
     }
printf("\nIs the plot to be printed? 1 Yes or 2 No   ");
  scanf("%s",plotorprint);
  if(plotorprint[0] == '1')
    {
      strcpy(systemtext,PRINTTEXT);
    }
  else
    {
      strcpy(systemtext,SCREENTEXT);
    }

/*This section initializes the variables which need to be*/

stbndx = 0;
datandx = 0;
optndx = 0;
A[BUFFSIZE] = 0;
strobecnt = 0;

/*This section reads in a buffer of the strobe file and calls
a subroutine 'findstrb' to locate the first strobe. If no
strobe values are found or if the file is empty, the
appropriate response is given and the program is exited. If a
strobe is found, 'nottop' is set to 0 to indicate that we are
at the top of the strobe. 'findstrb' also returns with the
```

Strobesum.c Program

```
/*This program inputs two files, a strobe file and a data
file, then produces an output summation file. The summation is
triggered by the strobe pulse in the strobe file and the
output is written from the addition buffer.  The program will
then plot the summed file to either the screen or the printer.
This program looks for a strobe that is unique.  The algorithm
is based on a strobe that is negative at 'off' state and then
goes to +100 or -100 within zero to show a strobe has occured.
The presence of a leading data point within this range
triggers the algorithm to start summing the data at the zero-
th element in the output buffer.
If a different strobe is to be used, it must first be clearly
identified as to its digitized values.  This program can then
be modified so that it will recognize the leading edge of the
strobe.*/

/*This section defines or delares all variables*/
#include <stdio.h>
#define FLBUFSIZE   125
#define BUFFSIZE     125
#define STRBBUFFSIZE FLBUFSIZE*2
#define MINUSONE    (-1)
#define PRINTTEXT  "plotp -YlPtempfile -xh100  -g lmxd\n"
#define SCREENTEXT "plotp -YlPtempfile -x100 -g lmmd\n"

char  systemtext[50];
char  BTOA[50] = {"btoa "};
char  plotorprint[2];
char  blanks[30] = {" "};
int  i,j,DONE,strndx,datandx,optndx;
int  nstrb_bytes,ndata_bytes;
int  A[BUFFSIZE] = 0;
short  strbbuff[STRBBUFFSIZE];
static short  databuf[FLBUFSIZE];
int  fpstrb,fpdata,fpout;
char  data_in_file[30],data_out_file[30];
char  strb_file_in[30];
short  *strbptr1, strbptr2;
int  nottop;  /*if a 1 at neg of strobe, 0 means at top of
                                                  strobe*/
int  strobecnt

main()
{
  while (DONE ==0)
    {
/*This section queries the operator as to which files should
be opened or created, and then opens or creates them.Error
```

```
        else
          control[7] = -minimum);
        maximum += control[7];
    /* scale data to y axis */
        if (maximum > yt )
          {
             control[9] = maximum/yt;  /* scale down */
             if ((maximu  % yt)>0) control[9] ++;
          }
        else
             control[8] = yt/maximum;  /* scale up 8 */
    }  /* end scale_data  */

    /*****************************************************************/

    int   findmax(gpbuff,count)
    short  *gpbuff;
    int   count;
     {
     short  shortmax;
    int   k,returnval;
        shortmax = gpbuff[0];
        for(k = 0; k < count; K++)
          {
             if(gpbuff[k] > shortmax)
               shortmax = gpbuff[k];
          }
        returnval = shortmax;
        return(returnval);
    }  /*  end findmax  */

    /*****************************************************************/

    int   findmin(gpbuff,count)
    short  *gpbuff;
    int   count;

    {
    short  shortmin;
    int  m,returnmim;
    shortmin = gpbuff[0];
        for(m=0;m<count;m++)
          {
             if(gpbuff[m] < shortmin)
               shortmin = gpbuff[m];
          }
        returnmim = shortmim;
        return(returnmim);

    }  /* end findmin */

    /***********************END OF FILE*********************************
```

APPENDIX F

MASSCOMP MC-500 Specifications and Capabilities

Specifications

Processing elements: A 10-MHz 68010 CPU and a 10-MHz 68000, the latter handling page faults in virtual memory while the former executes another program. The CPU has a 4-kbyte cache, a 1024-element translation buffer, a 1024-entry I/O map, and three serial lines.

An optional FP-501 floating point processor performs single- and double-precision arithmetic and holds 16 sets of 32 double-precision (64-bit) registers.

An optional AP-501 array processor executes single- and double-precision floating-point vectors, locally stores 32-bit elements, incorporates a 68000 that moves data to and from virtual memory, and requires the FP-510 for double-precision tasks.

Memory: 1 Mbyte (Minimum), 6 Mbytes (maximum); up to 16 Mbytes of virtual memory per process.

Mass storage: A 1-Mbyte floppy-disk drive, a 40-Mbyte (formatted) Winchester disk drive, and a 45-Mbyte ¼-in. tape cartridge (all optional).

Network: An optional plug-in Ethernet controller and supporting software.

Backplane: 8-slot enhanced Multibus with transfer rate of 6 Mbytes/s; eight alphanumerical terminals can be attached through backplane.

Ports: Three serial lines through CPU.

Graphics: Two 19-in. terminals (optional) attached through Multibus, each with its own 68000 and 128-kbyte program storage. The monochrome version displays 1024 by 800 pixels, the color unit 832 by 600 pixels by 6 or 10 bit planes. Each has 60-Hz noninterlaced refreshing and a 117-element keyboard; a mouse (or puck) and tablet are optional.

Operating system: The real-time RTU with virtual memory is compatible with the source code of Unix System III and of Berkeley 4.2 Unix and includes communications, text-processing, and support utilities; editors; source-code control system; and Bourne and C shells.

Programming languages: Fortran 77, C, and Pascal.

Minimum OEM configuration: CPU, power supply, 1 Mbyte of RAM, 8-slot backplane, and cabinet.

From Electronic Design, volume 32; July 26, 1984; page 112.

The following section is based on "Designing For High Performance Data Acquisition," Computer Design, September 1983.

MASSCOMP's MC-500 is a 32-bit Unix-based mini-computer system built for realtime scientific and technical applications. The system architecture distributes the processing load among three dedicated subsystems that perform high speed computation, graphics display, and data acquisition (Fig F-1).

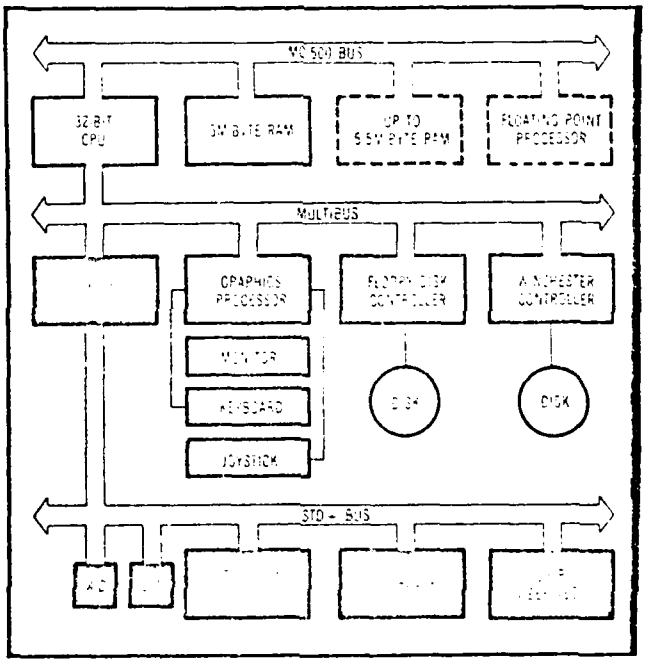


Figure F-1. MASSCOMP's MC-500 Architecture (5:2)

These subsystems are linked by three high speed buses: a proprietary bus links the main central processing unit (CPU) with physical memory storage; Intel's Multibus connects system peripherals to the graphics display processors; and twin STD buses channel data between the system and user input/output (I/O) devices.

The data acquisition and central processor (DA/CP) allows the MC-500 to collect realtime data. The DA/CP is a frontend, user-programmable, data acquisition controller and allows input or output analog or digital data at rates up to 2 M bytes/s. When coupled with an analog input interface, the DA/CP enables the minicomputer to acquire analog data at a 1M, 12-bit sample/s rate. The MC-500 also features a continuous 2M-bytes/s transfer rate from an analog to digital (A-D) converter to MC-500 memory.

Clock Control

The DA/CP uses 2901 bit-slice chips for its data path, and 2911 chips for its program sequencer. Usually, bit-slice based processors have wide instructions to simultaneously control the sequencer and the data path. Instructions ranging from 64 to over 100 bits are common. However, the DA/CP has a narrow instruction width of only 40 bits, and includes instructions for controlling the data path and the sequencer. This approach results in a compact instruction set. MOVE class instructions control the data path and cause the sequencer to increment the program counter. BRANCH class instructions control the sequencer in program jumps based on the previous MOVE instructions.

While the narrow instruction width saves program-store random access memory (RAM) space, this savings could easily be lost in decoding MOVE and BRANCH instructions from the same bits. The clocking scheme simplifies decoding and also reduces

gate delays in decoding instructions and achieves 125-ns execution times for all DA/CP instructions.

The DA/CP's clock control circuitry is shown in FigF-2(a), while Fig F-2(b) illustrates the specifics of clock timing. The

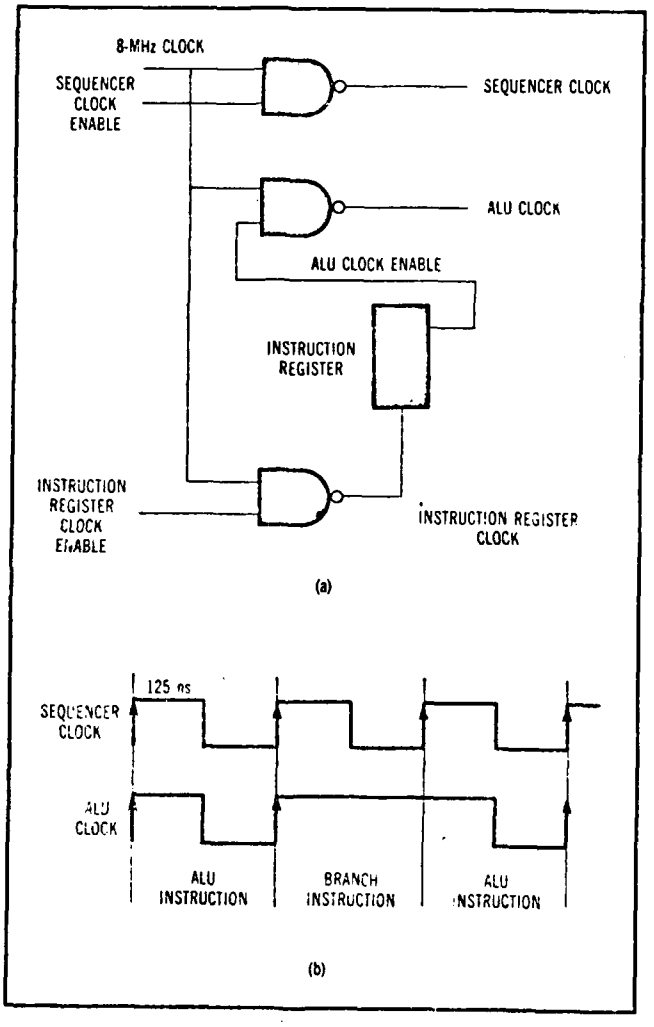


Figure F-2. Clock-control Circuitry (5:2)

arithmetic logic unit (ALU) clock enable signal is represented by 1 bit of the MOVE/BR instruction. When set, this bit enables the 2901's internal clock. The rest of the MOVE/BR

instruction bits control data paths.

Once the 2901's clock is enabled, bits representing the 2911's next address multiplex control signal are forced to an 11 logic state. This results in the selection of an internal program counter value as the next address. Other MOVE/BR instruction bits that could possibly affect program flow are now ignored. This instruction is known as a MOVE class instruction.

If the ALU clock enable bit is zero, the instruction is a BRANCH class instruction. Here the test condition programmable logic arrays (PAL's) determine the 2911's multiplexer controls, with the ALU clock to the 2901s disabled. Though instruction bits still affect the data path's combinational logic, the disabled clock inhibits writes to 2901 registers or the data store. Thus, the data path is effectively blocked during a BRANCH class instruction. Control fields for the sequencer and the data path can be overlaid in the same instruction bits without a lot of decode logic or associated gate delays.

The data path of the [illegible] class instruction is 24 bits wide. Each data-store location is divided into 3 bytes: high byte, mid byte, and low byte (Fig F-3). The MOVE class instructions allow operations on these data types. Byte data occupy the low byte of a data-store location. Words are in the mid and low bytes, and [illegible] (24 bit) use all 3 bytes of a data-store location. The [illegible] test conditions in the BRANCH class instruction for each of these data types.

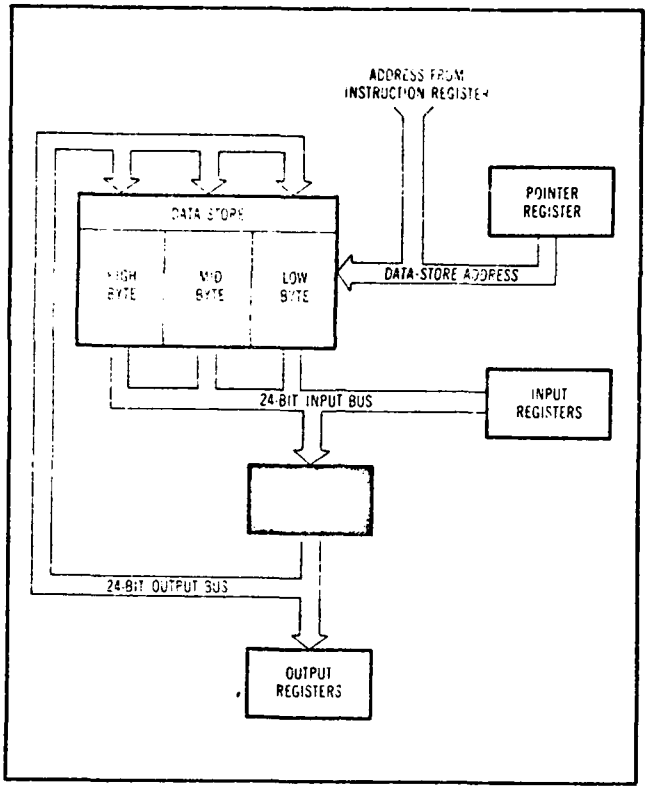


Figure F-3. Data Storage (5:3)

All MOVE class instructions allow merge operations on the input bus to the ALU. The 2-byte wide input registers can put data on low byte, mid byte, or both , while a data-store location provides the other input bus bytes. The merge operation is useful for assembling bytes from an STD bus module into words in the data-store before sending them to system memory. For example, all 12-bit A-Ds on the STD bus (exc Masscomp's) transfer data 1 byte at a time. An STD bus read operation can put 1 byte in the data store, and the next read can merge that byte with the rest of the data from the STD bus input register on the input bus.

All MOVE instructions can write selected bytes of a data store location while writing the output registers on the output bus. This feature improves the performance of some common DA/CP operations. For example, a Multibus direct memory access (DMA) address kept in the data store can be updated and returned to the data store in the same instruction that writes it to an output register for use in a DMA. The DA/CP also has shift and rotate operations on byte, word, and gulp data types.

Although all MOVE class instructions produce a 24-bit result in the ALU, test condition in the BRANCH instructions exist for the byte data, word data, and the full 24-bit gulp. Arithmetic tests fall into two groups: unsigned integer tests, and tests on 2's-complement signed data. Full 24-bit arithmetic is only done on Multibus addresses for DMA. Since these are not signed numbers, the tests provided are for unsigned integers. Word data can be unsigned integers, like those from a 16-bit digital I/O device, or 2's-complement data from a biploar A-D converter. Therefore, a set of signed and unsigned data tests are provided for word data. All the unsigned integer tests and several of the signed data tests are provided for byte data. BRANCH-IF-TRUE and BRANCH-IF-FALSE instructions exist for all the tests. There are 32-arithmetic test conditions in all.

Addressing

The DA/CP has a 1K-instruction program store and a separate,

256-location data store. Direct addressing allows a BRANCH class instruction to jump to any location in the program store and a MOVE instruction to access any location in the data store.

Moreover the DA/CP has an indirect addressing mode that uses the contents of a data store location as a BRANCH address. Changing the contents of this data-store location alters the program flow. This feature is useful for eliminating code from a program loop after it is no longer needed. For example, a routine can include code to test input data for a threshold crossing. Once the threshold is detected, changing the value in the data-store location that was used to jump to that location eliminates the threshold-detect code from the routine. This yields faster code than using a threshold-detect flag bit that is tested every pass through the routine to determine if the threshold-detect code should be invoked. In addition, DA/CP features indirect data addressing in the data store. A pointer register on the output bus can be loaded with a value and its contents can be used as the data store's address source in subsequent MOVE instructions. This feature is useful for implementing ring buffers in the data store. A block of data-store space is allocated for the buffer, and a load address and an unload address are also kept in the data store. When data are put into the ring buffer, the load pointer is moved into the pointer register and used to address the data store for the load. The address is incremented and saved for the next load. The same process is done

for unloading the ring buffer. For high speed data output devices, it is necessary to compensate for normal Multibus latency by prefetching some data into a ring buffer.

The DA/CP has also eliminated interrupt service overhead by overlapping it with useful instructions and maintaining an interrupt-disable default condition. This interrupt means no overhead penalty for multitasking data acquisition.

APPENDIX G

Integrated Circuit Specifications

OPERATIONAL AMPLIFIERS

MC1748G
MC1748CG

HIGH PERFORMANCE MONOLITHIC OPERATIONAL AMPLIFIER

. . . designed for use as a summing amplifier, integrator, or amplifier with operating characteristics as a function of the external feedback components.

- Noncompensated MC1741G
- Single 30 pF Capacitor Compensation Required For Unity Gain
- Short-Circuit Protection
- Offset Voltage Null Capability
- Wide Common-Mode and Differential Voltage Ranges
- Low-Power Consumption
- No Latch Up

OPERATIONAL AMPLIFIER
INTEGRATED CIRCUIT

MONOLITHIC
SILICON EPITAXIAL
PASSIVATED

METAL PACKAGE
CASE 601

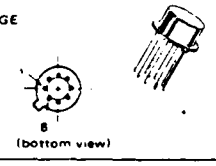


FIGURE 1 – POWER BANDWIDTH
(LARGE SIGNAL SWING versus FREQUENCY)

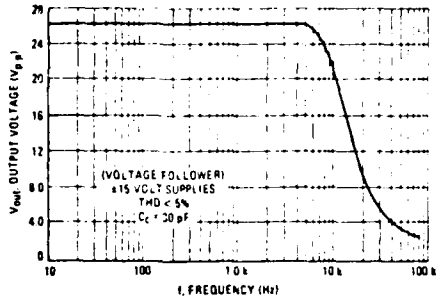


FIGURE 2 – OPEN LOOP FREQUENCY RESPONSE

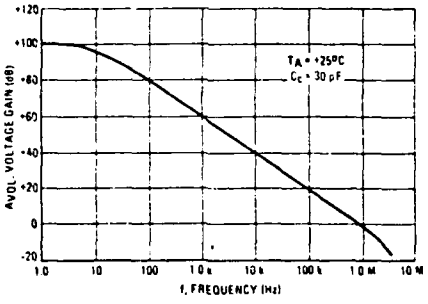


FIGURE 3 – CIRCUIT SCHEMATIC

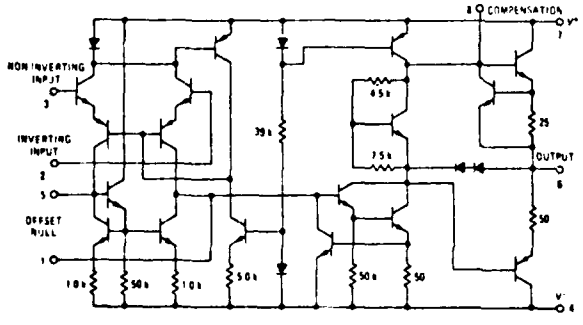
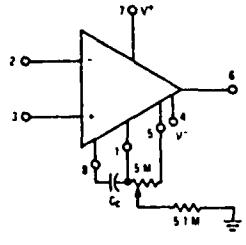


FIGURE 4 – OFFSET ADJUST AND FREQUENCY COMPENSATION



MAXIMUM RATINGS (T_A = +25°C unless otherwise noted)

Rating	Symbol	MC1748G	MC1748CG	Unit
Power Supply Voltage	V^+ V^-	+22 -22	+18 -18	Vdc
Differential Input Signal	V_{in}	±30		Volts
Common-Mode Input Swing ①	CMV_{in}	±15		Volts
Output Short Circuit Duration	t_S	Continuous		
Power Dissipation (Package Limitation) Derate above T_A = +25°C	P_D	680 4.6		mW mW/°C
Operating Temperature Range	T_A	-55 to +125	0 to +75	°C
Storage Temperature Range	T_{stg}	-65 to +150	-65 to +150	°C

ELECTRICAL CHARACTERISTICS (V^+ = +15 Vdc, V^- = -15 Vdc, T_A = +25°C unless otherwise noted)

Characteristics	Symbol	MC1748G			MC1748CG			Unit
		Min	Typ	Max	Min	Typ	Max	
Input Bias Current	I_b							μAdc
T_A = +25°C		–	0.08	0.5	–	0.08	0.5	
T_A = T_{low} to T_{high} ②		–	0.3	1.5	–	–	0.8	
Input Offset Current	$\|I_{io}\|$							μAdc
T_A = +25°C		–	0.02	0.2	–	0.02	0.2	
T_A = T_{low} to T_{high}		–	0.08	0.5	–	–	0.3	
Input Offset Voltage ($R_S \leq 10$ k Ω)	$\|V_{io}\|$							mVdc
T_A = +25°C		–	1.0	5.0	–	1.0	6.0	
T_A = T_{low} to T_{high}		–	–	6.0	–	–	7.5	
Differential Input Impedance (Open-Loop, f = 20 Hz)								
Parallel Input Resistance	R_p	0.3	2.0	–	0.3	2.0	–	Megohm
Parallel Input Capacitance	C_p	–	1.4	–	–	1.4	–	pF
Common-Mode Input Impedance (f = 20 Hz)	$Z_{(in)}$	–	200	–	–	200		Megohms
Common-Mode Input Voltage Swing	CMV_{in}	±12	±13	–	±12	±13	–	V_{pk}
Common-Mode Rejection Ratio (f = 100 Hz)	CM_{rej}	70	90	–	70	90	–	dB
Open-Loop Voltage Gain, (V_o = ±10 V, R_L = 2.0 k ohms)	A_{VOL}							V/V
T_A = +25°C		50,000	200,000	–	20,000	200,000	–	
T_A = T_{low} to T_{high}		25,000	–	–	15,000	–	–	
Step Response (V_{in} = 20 mV, C_c = 30 pF, R_L = 2 kΩ, C_L = 100 pF)								
Rise Time	t_r	–	0.3	–	–	0.3	–	μs
Overshoot Percentage		–	5.0	–	–	5.0	–	%
Slew Rate	dV_{out}/dt	–	0.8	–	–	0.8	–	V/μs
Output Impedance (f = 20 Hz)	Z_{out}	–	75	–	–	75	–	ohms
Short-Circuit Output Current	I_{SC}	–	25	–	–	25	–	mAdc
Output Voltage Swing (R_L = 10 k ohms)	V_o	±12	±14	–	±12	±14	–	Vpk
R_L = 2 k ohms (T_A = T_{low} to T_{high})		±10	±13	–	±10	±13	–	
Power Supply Sensitivity								μV/V
V^- = constant, $R_s \leq 10$ k ohms	S+	–	30	150	–	30	150	
V^+ = constant, $R_s \leq 10$ k ohms	S-	–	30	150	–	30	150	
Power Supply Current	I_D^+	–	1.67	2.83	–	1.67	2.83	mAdc
	I_D^-	–	1.67	2.83	–	1.67	2.83	
DC Quiescent Power Dissipation (V_o = 0)	P_D	–	50	85	–	50	85	mW

① For supply voltages less than ±15 V, the Maximum Input Voltage is equal to the Supply Voltage

② T_{low} 0°C for MC1748CG
-55°C for MC1748G
T_{high} +75°C for MC1748CG
+125°C for MC1748G

MC1741S
MC1741SC

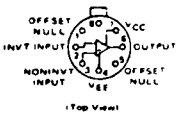
HIGH SLEW-RATE INTERNALLY-COMPENSATED OPERATIONAL AMPLIFIER

The MC1741S/MC1741SC is functionally equivalent, pin compatible, and possesses the same ease of use as the popular MC1741 circuit, yet offers 20 times higher slew rate and power bandwidth. This device is ideally suited for D-to-A converters due to its fast settling time and high slew rate.

- High Slew Rate – 10 V/μs Guaranteed Minimum (for unity gain only)
- No Frequency Compensation Required
- Short-Circuit Protection
- Offset Voltage Null Capability
- Wide Common-Mode and Differential Voltage Ranges
- Low Power Consumption
- No Latch-Up

OPERATIONAL AMPLIFIER
SILICON MONOLITHIC INTEGRATED CIRCUIT

G SUFFIX
METAL PACKAGE
CASE 601-02

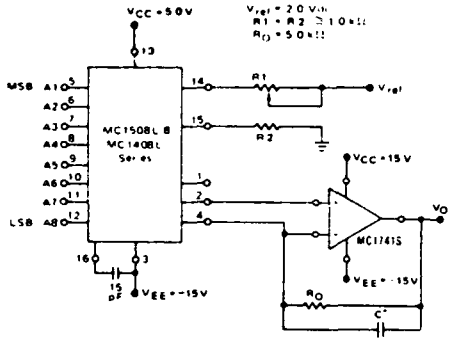


(Top View)

P1 SUFFIX
PLASTIC PACKAGE
CASE 626
(MC1741SC Only)

(Top View)

TYPICAL APPLICATION OF OUTPUT CURRENT TO VOLTAGE TRANSFORMATION FOR A D-TO-A CONVERTER



Pins not shown are not connected

Settling time to within 1/2 LSB (±19.5 mV) is approximately 4.0 μs from the time that all bits are switched.

*The value of C may be selected to minimize overshoot and ringing (C ≈ 150 pF)

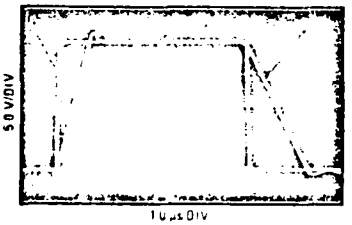
Theoretical V_O

$$V_O = \frac{V_{ref}}{R1}(R_O)\left[\frac{A1}{2}+\frac{A2}{4}+\frac{A3}{8}+\frac{A4}{16}+\frac{A5}{32}+\frac{A6}{64}+\frac{A7}{128}+\frac{A8}{256}\right]$$

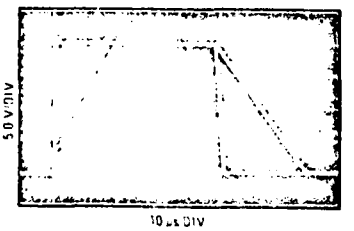
Adjust V_{ref}, R1 or R_O so that V_O with all digital inputs at high level is equal to 9.961 volts

$$V_O = \frac{2\ V}{1\ k}(5\ k)\left[\frac{1}{2}+\frac{1}{4}+\frac{1}{8}+\frac{1}{16}+\frac{1}{32}+\frac{1}{64}+\frac{1}{128}+\frac{1}{256}\right] = 10\ V\left[\frac{255}{256}\right] = 9.961\ V$$

MC1741S LARGE-SIGNAL TRANSIENT RESPONSE



STANDARD MC1741 versus MC1741S RESPONSE COMPARISON



See Packaging Information Section for outline dimensions

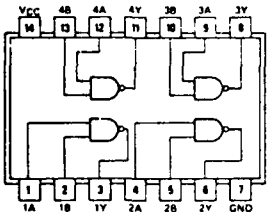
SSI GATES . . . LOGIC AND PIN ASSIGNMENTS (TOP VIEWS)

03

QUADRUPLE 2-INPUT POSITIVE-NAND GATES WITH OPEN-COLLECTOR OUTPUTS

positive logic:
$Y = \overline{AB}$

See page 88



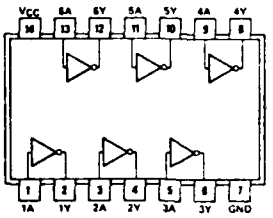
SN5403/SN7403(J, N)
SN54L03/SN74L03(J, N)
SN54LS03/SN74LS03(J, N, W)
SN54S03/SN74S03(J, N, W)

04

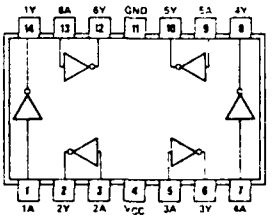
HEX INVERTERS

positive logic:
$Y = \overline{A}$

See page 86



SN5404/SN7404(J, N)
SN54H04/SN74H04(J, N)
SN54L04/SN74L04(J, N)
SN54LS04/SN74LS04(J, N, W)
SN54S04/SN74S04(J, N, W)



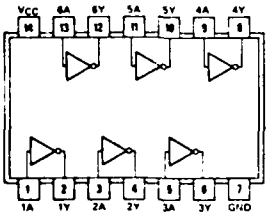
SN5404/SN7404(W)
SN54H04/SN74H04(W)
SN54L04/SN74L04(T)

05

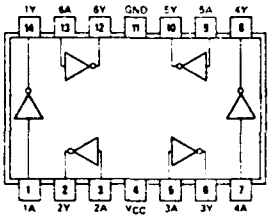
HEX INVERTERS WITH OPEN-COLLECTOR OUTPUTS

positive logic:
$Y = \overline{A}$

See page 88



SN5405/SN7405(J, N)
SN54H05/SN74H05(J, N)
SN54LS05/SN74LS05(J, N, W)
SN54S05/SN74S05(J, N, W)



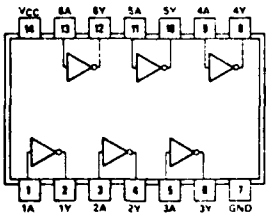
SN5405/SN7405(W)
SN54H05/SN74H05(W)

06

HEX INVERTER BUFFERS/DRIVERS WITH OPEN-COLLECTOR HIGH-VOLTAGE OUTPUTS

positive logic:
$Y = \overline{A}$

See page 10F



SN5406/SN7406(J, N, W)

recommended operating conditions

	54 FAMILY / 74 FAMILY	SERIES 54 SERIES 74 '00, '04, '10, '20, '30			SERIES 54H SERIES 74H 'H00, 'H04, 'H10, 'H20, 'H30			SERIES 54L SERIES 74L 'L00, 'L04, 'L10, 'L20, 'L30			SERIES 54LS SERIES 74LS 'LS00, 'LS04, 'LS10, 'LS20, 'LS30			SERIES 54S SERIES 74S 'S00, 'S04, 'S10, 'S20, 'S30, 'S133			UNIT
		MIN	NOM	MAX	MIN	NOM	MAX	MIN	NOM	MAX	MIN	NOM	MAX	MIN	NOM	MAX	
Supply voltage, V_{CC}	54 Family	4.5	5	5.5	4.5	5	5.5	4.5	5	5.5	4.5	5	5.5	4.5	5	5.5	V
	74 Family	4.75	5	5.25	4.75	5	5.25	4.75	5	5.25	4.75	5	5.25	4.75	5	5.25	
High-level output current, I_{OH}	54 Family			−400			−500			−100			−400			−1000	μA
	74 Family			−400			−500			−200			−400			−1000	
Low-level output current, I_{OL}	54 Family			16			20			2			4			20	mA
	74 Family			16			20			3.6			8			20	
Operating free-air temperature, T_A	54 Family	−55		125	−55		125	−55		125	−55		125	−55		125	°C
	74 Family	0		70	0		70	0		70	0		70	0		70	

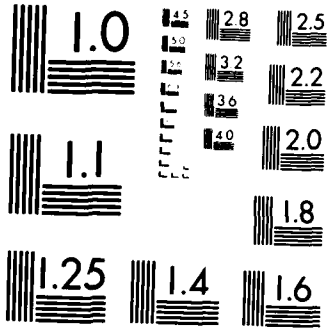
electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER		TEST FIGURE	TEST CONDITIONS†			SERIES 54 SERIES 74 '00, '04, '10, '20, '30			SERIES 54H SERIES 74H 'H00, 'H04, 'H10, 'H20, 'H30			SERIES 54L SERIES 74L 'L00, 'L04, 'L10, 'L20, 'L30			SERIES 54LS SERIES 74LS 'LS00, 'LS04, 'LS10, 'LS20, 'LS30			SERIES 54S SERIES 74S 'S00, 'S04, 'S10, 'S20, 'S30, 'S133			UNIT
						MIN	TYP‡	MAX	MIN	TYP‡	MAX	MIN	TYP‡	MAX	MIN	TYP‡	MAX	MIN	TYP‡	MAX	
V_{IH}	High-level input voltage	1, 2				2			2			2			2			2			V
V_{IL}	Low-level input voltage	1, 2			54 Family			0.8			0.8			0.7			0.7			0.8	V
					74 Family			0.8			0.8			0.7			0.8			0.8	
V_I	Input clamp voltage	3	V_{CC} = MIN,	I_I = §,				*−1.5			* 1.5						−1.5			−1.2	V
V_{OH}	High-level output voltage	1	V_{CC} = MIN, V_{IL} = V_{IL} max,		54 Family	2.4	3.4		2.4	3.5		2.4	3.3		2.5	3.4		2.5	3.4		V
			I_{OH} = MAX		74 Family	2.4	3.4		2.4	3.5		2.4	3.2		2.7	3.4		2.7	3.4		
V_{OL}	Low-level output voltage	2	V_{CC} = MIN, V_{IH} = 2 V,		54 Family		0.2	0.4		0.2	0.4		0.15	0.3		0.25	0.4			0.5	V
			I_{OL} = MAX		74 Family		0.2	0.4		0.2	0.4		0.2	0.4		0.35	0.5			0.5	
I_I	Input current at maximum input voltage	4	V_{CC} = MAX, V_I = 5.5 V					1			1			0.1			0.1			1	mA
I_{IH}	High-level input current	4	V_{CC} = MAX		V_{IH} = 2.4 V			40			50			10							μA
					V_{IH} = 2.7 V												20			50	
I_{IL}	Low-level input current	5	V_{CC} = MAX	V_{IL} = 0.3 V										−0.18							mA
				V_{IL} = 0.4 V	'LS30												−0.4				
					Others			−1.6			−2						−0.36				
				V_{IL} = 0.5 V																−2	
I_{OS}	Short-circuit output current¶	6	V_{CC} = MAX		54 Family	−20		−55	−40		−100	−3		−15	−6		−40	−40		−100	mA
					74 Family	−18		−55	−40		−100	−3		−15	−5		−42	−40		−100	
I_{CC}	Supply current	7	V_{CC} = MAX			See table on next page															mA

†For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions.
‡All typical values are at V_{CC} = 5 V, T_A = 25°C.
§I_I = −12 mA for SN54'/SN74', −8 mA for SN54H'/SN74H', and −18 mA for SN54LS'/SN74LS' and SN54S'/SN74S'.
¶Not more than one output should be shorted at a time, and for SN54H'/SN74H' and SN54S'/SN74S', duration of short-circuit should not exceed 1 second.
*The input clamp voltage specification is effective for Series 54/74 and 54H/74H parts date-coded 7332 or higher.

AD-A151 699 THE FIRST CORTICAL IMPLANT OF A SEMICONDUCTOR MULTIELECTRODE ARRAY: CONTI..(U) AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOOL OF ENGI.. 3/3

UNCLASSIFIED J D COLLIER ET AL. DEC 84 F/G 5/10 NL



supply current¶

TYPE	I_{CCH} (mA) Total with outputs high		I_{CCL} (mA) Total with outputs low		I_{CC} (mA) Average per gate (50% duty cycle)
	TYP	MAX	TYP	MAX	TYP
'00	4	8	12	22	2
'04	6	12	18	33	2
'10	3	6	9	16.5	2
'20	2	4	6	11	2
'30	1	2	3	6	2
'H00	10	16.8	26	40	4.5
'H04	16	26	40	58	4.5
'H10	7.5	12.6	19.5	30	4.5
'H20	5	8.4	13	20	4.5
'H30	2.5	4.2	6.5	10	4.5
'L00	0.44	0.8	1.16	2.04	0.20
'L04	0.66	1.2	1.74	3.06	0.20
'L10	0.33	0.6	0.87	1.53	0.20
'L20	0.22	0.4	0.58	1.02	0.20
SN54L30	0.11	0.33	0.29	0.51	0.20
SN74L30	0.11	0.2	0.29	0.51	0.20
'LS00	0.8	1.6	2.4	4.4	0.4
'LS04	1.2	2.4	3.6	6.6	0.4
'LS10	0.6	1.2	1.8	3.3	0.4
'LS20	0.4	0.8	1.2	2.2	0.4
'LS30	0.35	0.5	0.6	1.1	0.48
'S00	10	16	20	36	3.75
'S04	15	24	30	54	3.75
'S10	7.5	12	15	27	3.75
'S20	5	8	10	18	3.75
'S30	3	5	5.5	10	4.25
'S133	3	5	5.5	10	4.25

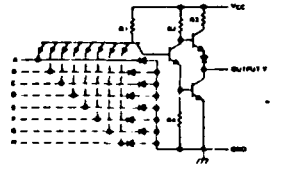
switching characteristics at V_{CC} = 5 V, T_A = 25°C

TYPE	TEST CONDITIONS#	t_{PLH} (ns) Propagation delay time, low-to-high-level output			t_{PHL} (ns) Propagation delay time, high-to-low-level output		
		MIN	TYP	MAX	MIN	TYP	MAX
'00, '10	C_L = 15 pF, R_L = 400 Ω		11	22		7	15
'04, '20			12	22		8	15
'30			13	22		8	15
'H00	C_L = 25 pF, R_L = 280 Ω		5.9	10		6.2	10
'H04			6	10		6.5	10
'H10			5.9	10		6.3	10
'H20			6	10		7	10
'H30			6.8	10		8.9	12
'L00, 'L04, 'L10, L20	C_L = 50 pF, R_L = 4 kΩ		35	60		31	60
'L30			35	60		70	100
'LS00, 'LS04 'LS10, 'LS20	C_L = 15 pF, R_L = 2 kΩ		9	20		10	20
'LS30			9	20		25	35
'S00, 'S04	C_L = 15 pF, R_L = 280 Ω	2	3	4.5	2	3	5
'S10, 'S20	C_L = 50 pF, R_L = 280 Ω		4.5			5	
'S30, 'S133	C_L = 15 pF, R_L = 280 Ω	2	4	6	2	4.5	7
	C_L = 50 pF, R_L = 280 Ω		5.5			6.5	

#Load circuits and voltage waveforms are shown on pages 148 and 149.

¶Maximum values of I_{CC} are over the recommended operating ranges of V_{CC} and T_A; typical values are at V_{CC} = 5 V, T_A = 25°C.

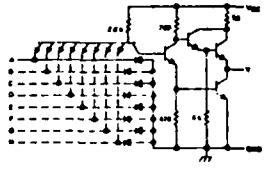
schematics (each gate)



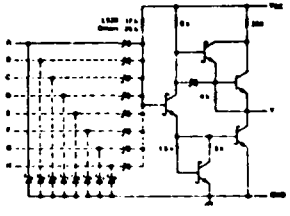
CIRCUIT	R1	R2	R3	R4
'00, '04, '10, '20, '30	4 k	1.6 k	130	1 k
L00, 'L04, 'L10, 'L20, 'L30	40 k	20 k	500	12 k

'00, '04, '10, '20, '30
'L00, 'L04, 'L10, 'L20, 'L30, CIRCUITS

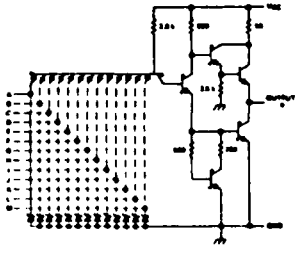
Input clamp diodes not on SN54L'/SN74L' circuits.



'H00, 'H04, 'H10, 'H20, 'H30 CIRCUITS



'LS00, 'LS04, 'LS10, 'LS20, 'LS30 CIRCUITS



'S00, 'S04, 'S10, 'S20, 'S30, 'S133 CIRCUITS

Resistor values shown are nominal and in ohms.

FLIP-FLOPS . . . LOGIC AND PIN ASSIGNMENTS (TOP VIEWS)

72

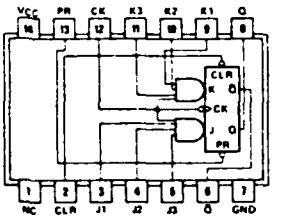
AND-GATED J-K MASTER-SLAVE FLIP-FLOPS WITH PRESET AND CLEAR

FUNCTION TABLE

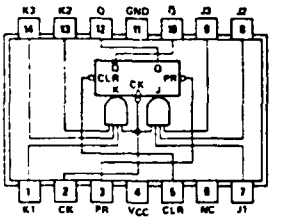
INPUTS					OUTPUTS	
PRESET	CLEAR	CLOCK	J	K	Q	$\bar{Q}$
L	H	X	X	X	H	L
H	L	X	X	X	L	H
L	L	X	X	X	H*	H*
H	H	⊓	L	L	Q_0	$\bar{Q}_0$
H	H	⊓	H	L	H	L
H	H	⊓	L	H	L	H
H	H	⊓	H	H	TOGGLE	

positive logic: J = J1·J2·J3, K1·K2·K3

See pages 120, 124, and 128



SN5472/SN7472(J, N)
SN54H72/SN74H72(J, N)
SN54L72/SN74L72(J, N)



SN5472/SN7472(W)
SN54H72/SN74H72(W)
SN54L72/SN74L72(T)

NC—No internal connection

73

DUAL J-K FLIP-FLOPS WITH CLEAR

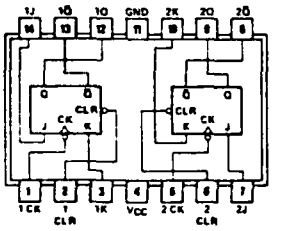
'73, 'H73, 'L73
FUNCTION TABLE

INPUTS				OUTPUTS	
CLEAR	CLOCK	J	K	Q	$\bar{Q}$
L	X	X	X	L	H
H	⊓	L	L	Q_0	$\bar{Q}_0$
H	⊓	H	L	H	L
H	⊓	L	H	L	H
H	⊓	H	H	TOGGLE	

'LS73
FUNCTION TABLE

INPUTS				OUTPUTS	
CLEAR	CLOCK	J	K	Q	$\bar{Q}$
L	X	X	X	L	H
H	↓	L	L	Q_0	$\bar{Q}_0$
H	↓	H	L	H	L
H	↓	L	H	L	H
H	↓	H	H	TOGGLE	
H	H	X	X	Q_0	$\bar{Q}_0$

See pages 120, 124, 128, and 130



SN5473/SN7473(J, N, W)
SN54H73/SN74H73(J, N, W)
SN54L73/SN74L73(J, N, T)
SN54LS73/SN74LS73(J, N, W)

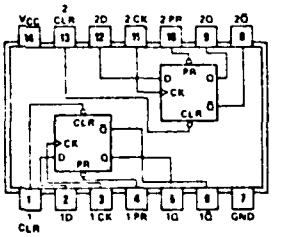
74

DUAL D-TYPE POSITIVE-EDGE-TRIGGERED FLIP-FLOPS WITH PRESET AND CLEAR

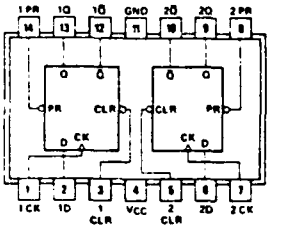
FUNCTION TABLE

INPUTS				OUTPUTS	
PRESET	CLEAR	CLOCK	D	Q	$\bar{Q}$
L	H	X	X	H	L
H	L	X	X	L	H
L	L	X	X	H*	H*
H	H	↑	H	H	L
H	H	↑	L	L	H
H	H	L	X	Q_0	$\bar{Q}_0$

See pages 120, 124, 128, 130, and 132



SN5474/SN7474(J, N)
SN54H74/SN74H74(J, N)
SN54L74/SN74L74(J, N)
SN54LS74/SN74LS74(J, N, W)
SN54S74/SN74S74(J, N, W)



SN5474/SN7474(W)
SN54H74/SN74H74(W)
SN54L74/SN74L74(T)

H = high level (steady state), L = low level (steady state), X = irrelevant

⊓ = high level pulse: data inputs should be held constant while clock is high; data is transferred to output on the falling edge of the pulse.

↑ = transition from low to high level, ↓ = transition from high to low level

Q_0 = the level of Q before the indicated input conditions were established.

TOGGLE: Each output changes to the complement of its previous level on each active transition (pulse) of the clock.

*This configuration is nonstable; that is, it will not persist when preset and clear inputs return to their inactive (high) level.

recommended operating conditions

		SERIES 54/74	'70			'72, '73, '76, '107			'74			'109			'110			'111			UNIT
			MIN	NOM	MAX	MIN	NOM	MAX	MIN	NOM	MAX	MIN	NOM	MAX	MIN	NOM	MAX	MIN	NOM	MAX	
Supply voltage, V_{CC}		Series 54	4.5	5	5.5	4.5	5	5.5	4.5	5	5.5	4.5	5	5.5	4.5	5	5.5	4.5	5	5.5	V
		Series 74	4.75	5	5.25	4.75	5	5.25	4.75	5	5.25	4.75	5	5.25	4.75	5	5.25	4.75	5	5.25	
High-level output current, I_{OH}					-400			-400			-400			-800			-800			-800	μA
Low-level output current, I_{OL}					16			16			16			16			16			16	mA
Pulse width, t_w	Clock high		20			20			30			20			25			25			ns
	Clock low		30			47			37			20			25			25			
	Preset or clear low		25			25			30			20			25			25			
Input setup time, t_{setup}			20↑			0↑			20↑			10↑			20↑			0↑			ns
Input hold time, t_{hold}			5↑			0↑			5↑			6↑			5↑			30↑			ns
Operating free-air temperature, T_A		Series 54	-55		125	-55		125	-55		125	-55		125	-55		125	-55		125	°C
		Series 74	0		70	0		70	0		70	0		70	0		70	0		70	

↑, ↓ The arrow indicates the edge of the clock pulse used for reference: ↑ for the rising edge, ↓ for the falling edge

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER		TEST CONDITIONS†	'70			'72, '73, '76, '107			'74			'109			'110			'111			UNIT
			MIN	TYP‡	MAX	MIN	TYP‡	MAX	MIN	TYP‡	MAX	MIN	TYP‡	MAX	MIN	TYP‡	MAX	MIN	TYP‡	MAX	
V_{IH}	High-level input voltage		2			2			2			2			2			2			V
V_{IL}	Low-level input voltage				0.8			0.8			0.8			0.8			0.8			0.8	V
V_I	Input clamp voltage	V_{CC} = MIN, I_I = -12 mA			-1.5			-1.5			-1.5			-1.5			-1.5			-1.5	V
V_{OH}	High-level output voltage	V_{CC} = MIN, V_{IH} = 2 V, V_{IL} = 0.8 V, I_{OH} = MAX	2.4	3.4		2.4	3.4		2.4	3.4		2.4	3.4		2.4	3.4		2.4	3.4		V
V_{OL}	Low-level output voltage	V_{CC} = MIN, V_{IH} = 2 V, V_{IL} = 0.8 V, I_{OL} = 16 mA		0.2	0.4		0.2	0.4		0.2	0.4		0.2	0.4		0.2	0.4		0.2	0.4	V
I_I	Input current at maximum input voltage	V_{CC} = MAX, V_I = 5.5 V			1			1			1			1			1			1	mA
I_{IH}	High-level input current	D, J, K, or $\overline{R}$	V_{CC} = MAX, V_I = 2.4 V		40			40			40			40			40			40	μA
		Clear			80			80			120			160			160			80	
		Preset			80			80			80			80			160			80	
		Clock			40			80			80			80			40			120	
I_{IL}	Low-level input current	D, J, K, or $\overline{R}$	V_{CC} = MAX, V_I = 0.4 V		-1.6			-1.6			-1.6			-1.6			-1.6			-1.6	mA
		Clear			-3.2			-3.2			-3.2			-4.8			-3.2			-3.2	
		Preset			-3.2			-3.2			-1.6			-3.2			-3.2			-3.2	
		Clock			-1.6			-3.2			-3.2			-3.2			-1.6			-4.8	
I_{OS}	Short-circuit output current§	Series 54	V_{CC} = MAX	-20	-57	-20	-57	-20	-57	-20	-57	-30	-85	-20	-57	-20	-57	-20	-57	mA
		Series 74		-18	-57	-18	-57	-18	-57	-18	-57	-30	-85	-18	-57	-18	-57	-18	-57	
I_{CC}	Supply current (Average per flip-flop)	V_{CC} = MAX, See Note 1		13	26		10	20		8.5	15		9	15		20	34		14	20.5	mA

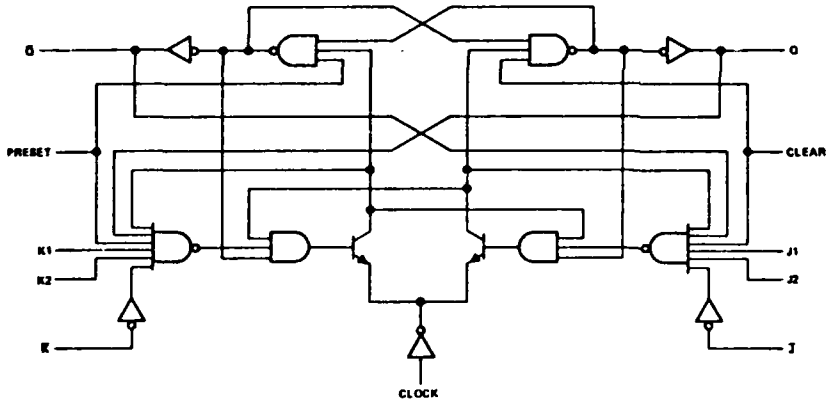
†For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions.
‡All typical values are at V_{CC} = 5 V, T_A = 25°C.
§Not more than one output should be shorted at a time.
NOTE 1: With all outputs open, I_{CC} is measured with the Q and $\overline{Q}$ outputs high in turn. At the time of measurement, the clock input is at 4.5 V for the '70, '110, and '111, and is grounded for all the others.
* The input clamp voltage specification is effective for Series 54/74 parts date coded 7332 or higher.

switching characteristics, V_{CC} = 5 V, T_A = 25°C

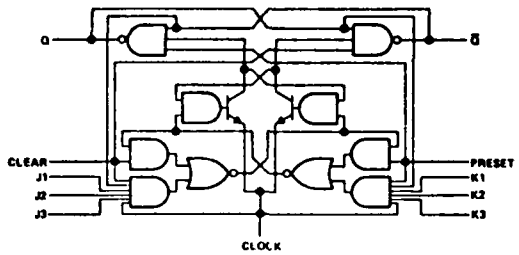
PARAMETER¶	FROM (INPUT)	TO (OUTPUT)	TEST CONDITIONS	'70			'72, '73 '76, '107			'74			'109			'110			'111			UNIT
				MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	
f_{max}				20	35		15	20		15	25		25	33		20	25		20	25		MHz
t_{PLH}	Preset	Q				50		16	25			25		10	15		12	20		12	18	ns
t_{PHL}	(as applicable)	$\bar{Q}$	C_L = 15 pF,			50		25	40			40		23	35		18	25		21	30	
t_{PLH}	Clear	$\bar{Q}$	R_L = 400 Ω,			50		16	25			25		10	15		12	20		12	18	ns
t_{PHL}	(as applicable)	Q	See Note 2			50		25	40			40		17	25		18	25		21	30	
t_{PLH}	Clock	Q or $\bar{Q}$		10	27	50	10	16	25	10	14	25	4	10	16	10	20	30	6	12	17	ns
t_{PHL}				10	18	50	10	25	40	10	20	40	9	18	28	6	13	20	10	20	30	

¶ f_{max} = maximum clock frequency, t_{PLH} = propagation delay time, low to high level output, t_{PHL} = propagation delay time, high to low level output.
NOTE 2: Load circuit and voltage waveforms are shown on page 148.

functional block diagrams



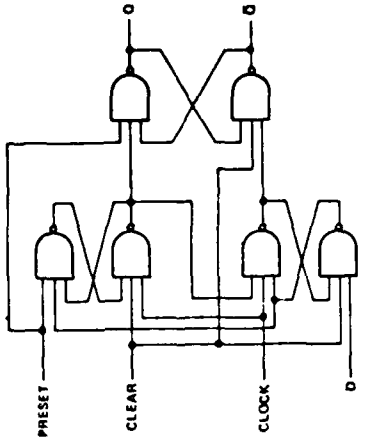
'70—GATED J-K WITH CLEAR AND PRESET



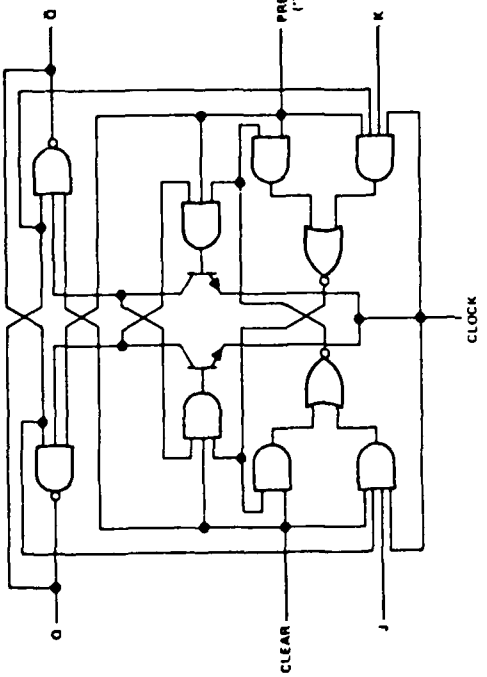
'72—GATED J-K WITH CLEAR AND PRESET

See following pages for:
'73—DUAL J-K WITH CLEAR
'74—DUAL D WITH CLEAR AND PRESET
'76—DUAL J-K WITH CLEAR AND PRESET
'107—DUAL J-K WITH CLEAR
'109—DUAL J-$\bar{K}$ WITH CLEAR AND PRESET
'110—GATED J-K WITH CLEAR AND PRESET
'111—DUAL J-K WITH CLEAR AND PRESET

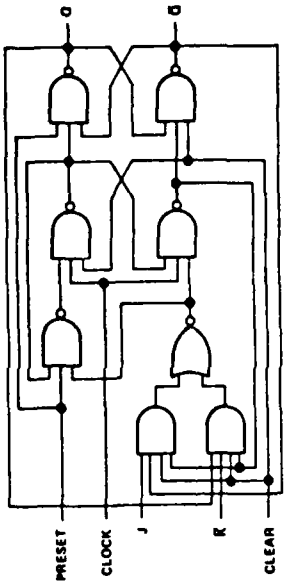
functional block diagrams (continued)



'74—DUAL D WITH CLEAR AND PRESET



'73—DUAL J-K WITH CLEAR
'76—DUAL J-K WITH CLEAR AND PRESET
'107—DUAL J-K WITH CLEAR



'109—DUAL J-K̄ WITH CLEAR AND PRESET

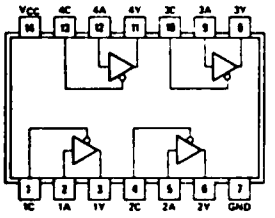
SSI GATES . . . LOGIC AND PIN ASSIGNMENTS (TOP VIEWS)

125

QUADRUPLE BUS BUFFER GATES WITH THREE-STATE OUTPUTS

positive logic:
Y = A
Output is off (disabled) when C is high.

See page 142



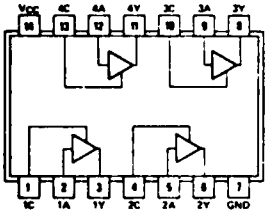
SN54125/SN74125(J, N, W)

126

QUADRUPLE BUS BUFFER GATES WITH THREE-STATE OUTPUTS

positive logic:
Y = A
Output is off (disabled) when C is low.

See page 142



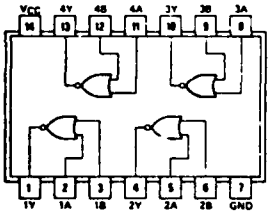
SN54126/SN74126(J, N, W)

128

SN54128 . . . 75-OHM LINE DRIVER
SN74128 . . . 50-OHM LINE DRIVER

positive logic:
$Y = \overline{A+B}$

See page 104



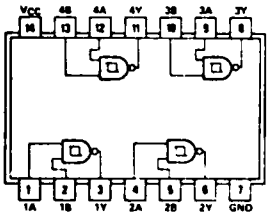
SN54128/SN74128(J, N, W)

132

QUADRUPLE 2-INPUT POSITIVE-NAND SCHMITT TRIGGERS

positive logic:
$Y = \overline{AB}$

See page 98



SN54132/SN74132(J, N, W)
SN54S132/SN74S132(J, N, W)

recommended operating conditions

		SERIES 54 SERIES 74 '125, '126			SERIES 54S SERIES 74S 'S134			UNIT
		MIN	NOM	MAX	MIN	NOM	MAX	
Supply voltage, V_{CC}	54 Family	4.5	5	5.5	4.5	5	5.5	V
	74 Family	4.75	5	5.25	4.75	5	5.25	
High-level output current, I_{OH}	54 Family			−2			−2	mA
	74 Family			−5.2			−6.5	
Low-level output current, I_{OL}				16			20	mA
Operating free-air temperature, T_A	54 Family	−55		125	−55		125	°C
	74 Family	0		70	0		70	

supply current¶

TYPE	TEST CONDITIONS: DATA INPUTS	TEST CONDITIONS: OUTPUT CONTROLS	I_{CC} (mA): MIN	TYP	MAX
'125	0 V	4.5 V		32	54
'126	0 V	0 V		36	62
'S134	0 V	0 V		7	13
	5 V	0 V		9	16
	5 V	5 V		14	25

¶Maximum values of I_{CC} are over the recommended operating ranges of V_{CC} and T_A; typical values are at V_{CC} = 5 V, T_A = 25°C.

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER		TEST FIGURE	TEST CONDITIONS†		SERIES 54 SERIES 74 '125, '126			SERIES 54S SERIES 74S 'S134			UNIT
					MIN	TYP‡	MAX	MIN	TYP‡	MAX	
V_{IH}	High-level input voltage	1, 2			2			2			V
V_{IL}	Low-level input voltage	1, 2					0.8			0.8	V
V_I	Input clamp voltage	3	V_{CC} = MIN,	I_I = §			−1.5			−1.2	V
V_{OH}	High-level output voltage	1	V_{CC} = MIN, V_{IH} = 2 V, V_{IL} = 0.8 V, I_{OH} = MAX	54 Family	2.4	3.3		2.4	3.4		V
				74 Family	2.4	3.1		2.4	3.2		
V_{OL}	Low-level output voltage	2	V_{CC} = MIN, V_{IH} = 2 V, V_{IL} = 0.8 V,	I_{OL} = MAX			0.4			0.5	V
$I_{O(off)}$	Off-state (high-impedance state) output current	19	V_{CC} = MAX, V_{IH} = 2 V, V_{IL} = 0.8 V	V_O = 2.4 V			40			50	μA
				V_O = 0.4 V			−40				
				V_O = 0.5 V						−50	
I_I	Input current at maximum input voltage	4	V_{CC} = MAX,	V_I = 5.5 V			1			1	mA
I_{IH}	High-level input current	4	V_{CC} = MAX	V_{IH} = 2.4 V			40				μA
				V_{IH} = 2.7 V						50	
I_{IL}	Low-level input current	5	V_{CC} = MAX	V_{IL} = 0.4 V			−1.6				mA
				V_{IL} = 0.5 V						−2	
I_{OS}	Short-circuit output current♦	6	V_{CC} = MAX	54 Family	−30	−70		−40	−70	−100	mA
				74 Family	−28	−70		−40	−70	−100	
I_{CC}	Supply current	7	V_{CC} = MAX		See table						mA

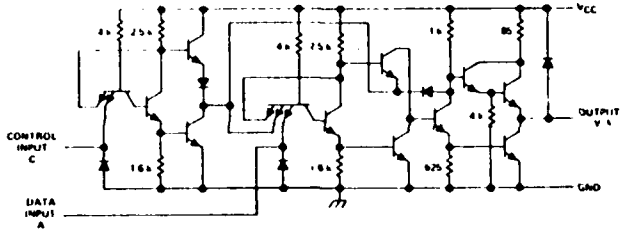
†For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions.
‡All typical values are at V_{CC} = 5 V, T_A = 25°C.
§I_I = −12 mA for SN54'/SN74' and −18 mA for SN54S'/SN74S'.
♦Not more than one output should be shorted at a time.

switching characteristics, V_{CC} = 5 V, T_A = 25°C

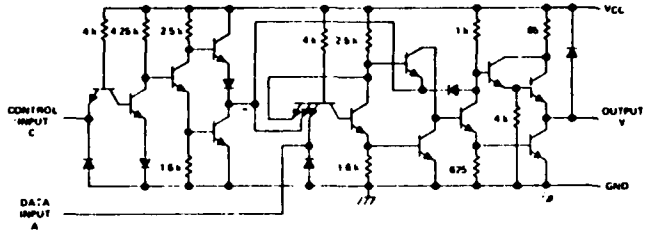
PARAMETER		SERIES 54/74						SERIES 54S/74S				UNIT
	TEST CONDITIONS#	'125			'126			TEST CONDITIONS#	'S134			
		MIN	TYP	MAX	MIN	TYP	MAX		MIN	TYP	MAX	
t_{PLH} Propagation delay time, low-to-high-level output	C_L = 50 pF, R_L = 400 Ω		8	13		8	13	C_L = 15 pF, R_L = 280 Ω	2	4	6	ns
								C_L = 50 pF, R_L = 280 Ω		5.5		
t_{PHL} Propagation delay time, high-to-low-level output			12	18		12	18	C_L = 15 pF, R_L = 280 Ω	2	5	7.5	ns
								C_L = 50 pF, R_L = 280 Ω		7		
t_{ZH} Output enable time to high level			11	17		11	18	C_L = 50 pF, R_L = 280 Ω		13	19.5	ns
t_{ZL} Output enable time to low level			16	25		16	25			14	21	ns
t_{HZ} Output disable time from high level	C_L = 5 pF, R_L = 400 Ω		5	8		10	16	C_L = 5 pF, R_L = 280 Ω		5.5	8.5	ns
t_{LZ} Output disable time from low level			7	12		12	18			9	14	ns

#Load circuit and voltage waveforms are shown on page 148.

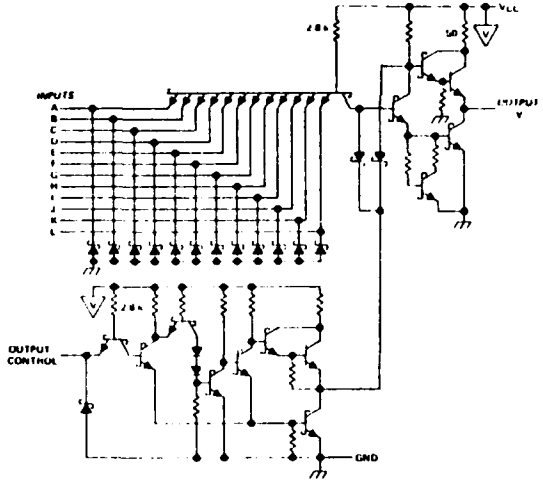
schematics (each gate)



'125 CIRCUITS



'126 CIRCUITS



'S134 CIRCUITS

Resistor values shown are nominal and in ohms.

MONOSTABLE MULTIVIBRATORS . . . LOGIC AND PIN ASSIGNMENTS (TOP VIEWS)

MONOSTABLE MULTIVIBRATORS

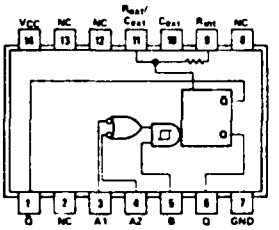
121

FUNCTION TABLE

INPUTS			OUTPUTS	
A1	A2	B	Q	$\bar{Q}$
L	X	H	L	H
X	L	H	L	H
X	X	L	L	H
H	H	X	L	H
H	↓	H	⊓	⊔
↓	H	H	⊓	⊔
↓	↓	H	⊓	⊔
L	X	↑	⊓	⊔
X	L	↑	⊓	⊔

See page 134

See Notes



SN54121/SN74121(J, N, W)
SN54L121/SN74L121(J, N, T)
'121 . . . R_{int} = 2 kΩ NOM
'L121 . . . R_{int} = 4 kΩ NOM

NC—No internal connection

RETRIGGERABLE MONOSTABLE MULTIVIBRATORS WITH CLEAR

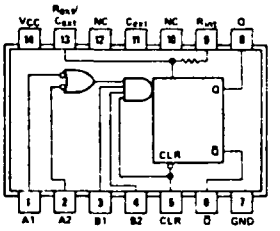
122

FUNCTION TABLE

INPUTS					OUTPUTS	
CLEAR	A1	A2	B1	B2	Q	$\bar{Q}$
L	X	X	X	X	L	H
X	H	H	X	X	L	H
X	X	X	L	X	L	H
X	X	X	X	L	L	H
X	L	X	H	H	L	H
H	L	X	↑	H	⊓	⊔
H	L	X	H	↑	⊓	⊔
H	X	L	H	H	L	H
H	X	L	↑	H	⊓	⊔
H	X	L	H	↑	⊓	⊔
H	H	↓	H	H	⊓	⊔
H	↓	↓	H	H	⊓	⊔
H	↓	H	H	H	⊓	⊔
↑	L	X	H	H	⊓	⊔
↑	X	L	H	H	⊓	⊔

See page 138

See Notes



SN54122/SN74122(J, N, W)
SN54L122/SN74L122(J, N, T)
'122 . . . R_{int} = 10 kΩ NOM
'L122 . . . R_{int} = 20 kΩ NOM

NC—No internal connection

DUAL RETRIGGERABLE MONOSTABLE MULTIVIBRATORS WITH CLEAR

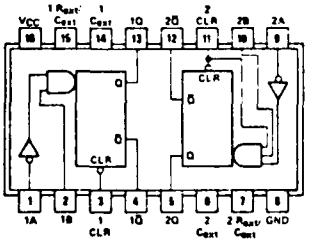
123

FUNCTION TABLE

INPUTS			OUTPUTS	
CLEAR	A	B	Q	$\bar{Q}$
L	X	X	L	H
X	H	X	L	H
X	X	L	L	H
H	L	↑	⊓	⊔
H	↓	H	⊓	⊔
↑	L	H	⊓	⊔

See page 138

See Notes



SN54123/SN74123(J, N, W)
SN54L123/SN74L123(J, N)

NOTES: A. H = high level (steady state), L = low level (steady state), ↑ = transition from low to high level, ↓ = transition from high to low level, ⊓ = one high-level pulse, ⊔ = one low-level pulse, X = irrelevant (any input, including transitions).

B. To use the internal timing resistor of '121, 'L121, '122, or 'L122, connect R_{int} to V_{CC}.

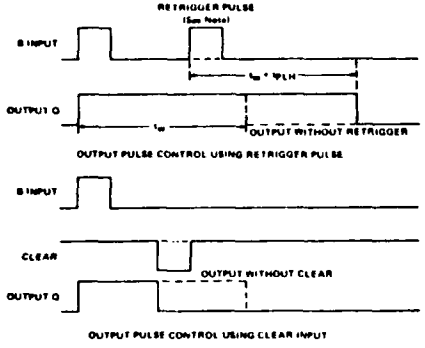
C. An external timing capacitor may be connected between C_{ext} and R_{ext}/C_{ext} (positive).

D. *For accurate repeatable pulse widths, connect an external resistor between R_{ext}/C_{ext} and V_{CC} with R_{int} open-circuited.*

E. To obtain variable pulse widths, connect external variable resistance between R_{int} or R_{ext}/C_{ext} and V_{CC}.

description

The '122, '123, 'L122, and 'L123 multivibrators feature d-c triggering from gated low-level-active (A) and high-level-active (B) inputs, and also provide overriding direct clear inputs. Complementary outputs are provided. The retrigger capability simplifies the generation of output pulses of extremely long duration. By triggering the input before the output pulse is terminated, the output pulse may be extended. The overriding clear capability permits any output pulse to be terminated at a predetermined time independently of the timing components R and C. Figure A below illustrates triggering the one-shot with the high-level-active (B) inputs.



NOTE: Retrigger pulse must not start before 0.22 C_{ext} (in picofarads) nanoseconds after previous trigger pulse.

FIGURE A—TYPICAL INPUT/OUTPUT PULSES

These monostables are designed to provide the system designer with complete flexibility in controlling the pulse width, either to lengthen the pulse by retriggering, or to shorten by clearing. The '122 and 'L122 each has an internal timing resistor which allows the circuit to be operated with only an external capacitor, if so desired. Applications requiring more precise pulse widths (up to 28 seconds) and not requiring the clear feature can best be satisfied with '121 or 'L121.

The output pulse is primarily a function of the external capacitor and resistor. For $C_{ext} > 1000$ pF, the output pulse width (t_w) is defined as:

$$t_w = K \cdot R_T \cdot C_{ext} \left(1 + \frac{0.7}{R_T}\right)$$

where

R_T is in kΩ (either internal or external timing resistor),
C_{ext} is in pF,
t_w is in ns,
K is 0.32 for '122, 0.28 for '123, 0.37 for 'L122, 0.33 for 'L123.

For pulse widths when $C_{ext} \leq 1000$ pF, see Figures B and C.

'122, '123
TYPICAL OUTPUT PULSE WIDTH
vs
EXTERNAL TIMING CAPACITANCE

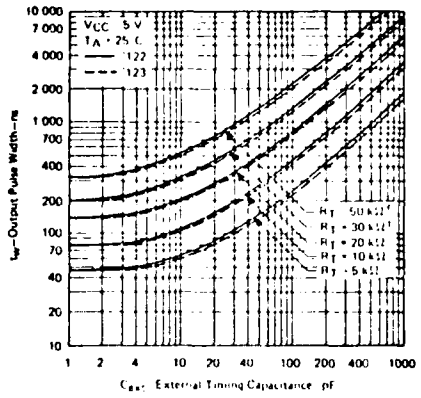


FIGURE B

'L122
TYPICAL OUTPUT PULSE WIDTH
vs
EXTERNAL TIMING CAPACITANCE

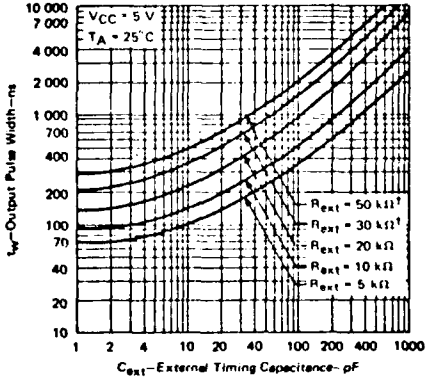


FIGURE C

†These values of resistance exceed the maximum recommended for use over the full temperature range of the SN54' and SN54L' circuits

recommended operating conditions

		54 FAMILY 74 FAMILY	SERIES 54 SERIES 74 '122, '123			SERIES 54L SERIES 74L 'L122, 'L123			UNIT
			MIN	NOM	MAX	MIN	NOM	MAX	
Supply voltage, V_{CC}		54 Family	4.5	5	5.5	4.5	5	5.5	V
		74 Family	4.75	5	5.25	4.75	5	5.25	
High-level output current, I_{OH}					-800			-400	μA
Low-level output current, I_{OL}					16			8	mA
Pulse width, t_w	A or B inputs high		40			50			ns
	A or B inputs low		40			50			
	Clear low		40			50			
External timing resistance, R_{ext}		54 Family	5		25	5		25	kΩ
		74 Family	5		50	5		50	
External capacitance, C_{ext}			No restriction			No restriction			
Wiring capacitance at R_{ext}/C_{ext} terminal					50			50	pF
Operating free-air temperature, T_A		54 Family	-55		125	-55		125	°C
		74 Family	0		70	0		70	

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER		TEST CONDITIONS†	SERIES 54 SERIES 74 122, '123			SERIES 54L SERIES 74L 'L122, 'L123			UNIT
			MIN	TYP‡	MAX	MIN	TYP‡	MAX	
V_{IH} High-level input voltage			2			2			V
V_{IL} Low-level input voltage					0.8			0.8	V
V_I Input clamp voltage		V_{CC} = MIN, I_I = -12 mA			-1.5			-1.5	V
V_{OH} High-level output voltage		V_{CC} = MIN, I_{OH} = MAX, See Note 1	2.4	3.4		2.4	3.4		V
V_{OL} Low-level output voltage		V_{CC} = MIN, I_{OL} = MAX, See Note 1		0.2	0.4		0.2	0.4	V
I_I Input current at maximum input voltage		V_{CC} = MAX, V_I = 5.5 V			1			1	mA
I_{IH} High-level input current	Data inputs	V_{CC} = MAX, V_I = 2.4 V			40			20	μA
	Clear input				80			40	
I_{IL} Low-level input current	Data inputs	V_{CC} = MAX, V_I = 0.4 V			-1.6			-0.8	mA
	Clear input				-3.2			-1.6	
I_{OS} Short-circuit output current♦		V_{CC} = MAX, See Note 1	-10		-40	-5		-20	mA
I_{CC} Supply current (quiescent or triggered)	'122, 'L122	V_{CC} = MAX, See Notes 2 and 3		23	28		11	14	mA
	'123, 'L123			46	66		23	33	

†For conditions shown as MIN or MAX, use the value specified under recommended operating conditions.
‡All typical values are at V_{CC} = 5 V, T_A = 25°C.
♦Not more than one output should be shorted at a time.
NOTES: 1. Ground C_{ext} to measure V_{OH} at Q, V_{OL} at $\overline{Q}$, or I_{OS} at Q. C_{ext} is open to measure V_{OH} at $\overline{Q}$, V_{OL} at Q, or I_{OS} at $\overline{Q}$.
2. Quiescent I_{CC} is measured (after clearing) with 2.4 V applied to all clear and A inputs, B inputs grounded, all outputs open, C_{ext} = 0.02 μF, and R_{ext} = 25 kΩ. R_{int} of '122 or 'L122 is open.
3. I_{CC} is measured in the triggered state with 2.4 V applied to all clear and B inputs, A inputs grounded, all outputs open, C_{ext} = 0.02 μF, and R_{ext} = 25 kΩ. R_{int} of '122 or 'L122 is open.

switching characteristics, V_{CC} = 5 V, T_A = 25°C, see note 4

PARAMETER†	FROM (INPUT)	TO (OUTPUT)	'122, '123 TEST CONDITIONS	'122, '123 MIN	'122, '123 TYP	'122, '123 MAX	'L122, 'L123 TEST CONDITIONS	'L122, 'L123 MIN	'L122, 'L123 TYP	'L122, 'L123 MAX	UNIT
t_{PLH}	A	Q	C_{ext} = 0, R_{ext} = 5 kΩ, C_L = 15 pF, R_L = 400 Ω		22	23	C_{ext} = 0, R_{ext} = 5 kΩ, C_L = 15 pF, R_L = 800 Ω		44	66	ns
	B				19	28			38	56	
t_{PHL}	A	$\overline{Q}$			30	40			60	80	ns
	B				27	36			54	72	
t_{PHL}	Clear	Q			18	27			36	54	ns
t_{PLH}		$\overline{Q}$			30	40			60	80	
t_{wQ} (min)	A or B	Q			45	65			90•	130•	ns
t_{wQ}	A or B	Q	C_{ext} = 1000 pF, R_{ext} = 10 kΩ, C_L = 15 pF, R_L = 400 Ω	'122: 3.08	3.42	3.76	C_{ext} = 400 pF, R_{ext} = 10 kΩ, C_L = 15 pF, R_L = 800 Ω	'L122: 1.7	1.9	2.1	μs
				'123: 2.76	3.03	3.37		'L123: 1.5•		2.1•	

†t_{PLH} = propagation delay time, low to high level output
t_{PHL} = propagation delay time, high to low level output
t_{wQ} = width of pulse at output Q
•These values for 'L123 are tentative.
NOTE 4: Load circuit and voltage waveforms are shown on page 148

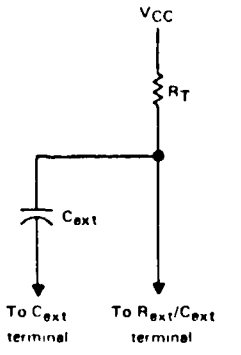
TYPICAL APPLICATION DATA

To prevent reverse voltage across C_{ext}, it is recommended that the method shown in Figure E be employed when using electrolytic capacitors and in applications utilizing the clear function. In all applications using the diode, the pulse width is:

$$t_w = K_D \cdot R_{ext} \cdot C_{ext} \left(1 + \frac{0.7}{R_{ext}}\right)$$

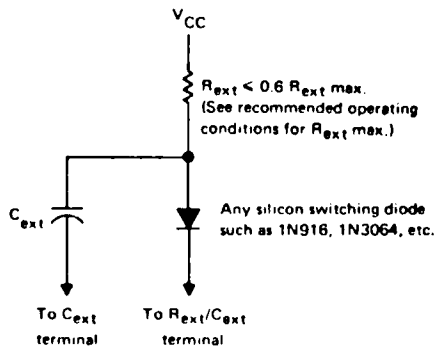
where

R_{ext} is in kΩ,
C_{ext} is in pF,
t_w is in ns,
K_D is 0.28 for '122, 0.25 for '123, 0.33 for 'L122, and 0.29 for 'L123.



TIMING COMPONENT CONNECTIONS WHEN C_{ext} ≤ 1000 pF

FIGURE D



TIMING COMPONENT CONNECTIONS WHEN C_{ext} > 1000 pF AND CLEAR IS USED

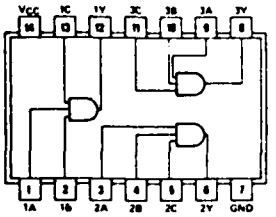
FIGURE E

SSI GATES . . . LOGIC AND PIN ASSIGNMENTS (TOP VIEWS)

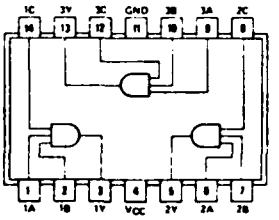
11
TRIPLE 3-INPUT POSITIVE-AND GATES

positive logic:
$Y = ABC$

See page 94



SN54H11/SN74H11(J, N)
SN54LS11/SN74LS11(J, N, W)
SN54S11/SN74S11(J, N, W)

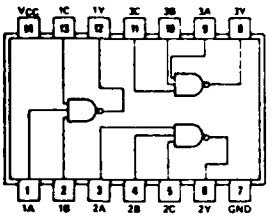


SN54H11/SN74H11(W)

12
TRIPLE 3-INPUT POSITIVE-NAND GATES WITH OPEN-COLLECTOR OUTPUTS

positive logic:
$Y = \overline{ABC}$

See page 88

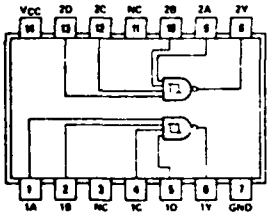


SN5412/SN7412(J, N, W)

13
DUAL 4-INPUT POSITIVE-NAND SCHMITT TRIGGERS

positive logic:
$Y = \overline{ABCD}$

See page 98



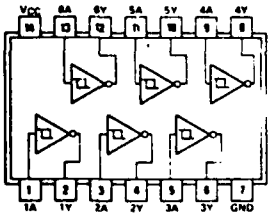
SN5413/SN7413(J, N, W)

NC—No internal connection

14
HEX SCHMITT-TRIGGER INVERTERS

positive logic:
$Y = \overline{A}$

See page 98



SN5414/SN7414(J, N, W)

recommended operating conditions

	54 FAMILY / 74 FAMILY	SERIES 54 / SERIES 74 '13			SERIES 54 / SERIES 74 '14			SERIES 54 / SERIES 74 '132			SERIES 54S / SERIES 74S 'S132			UNIT
		MIN	NOM	MAX	MIN	NOM	MAX	MIN	NOM	MAX	MIN	NOM	MAX	
Supply voltage, V_{CC}	54 Family	4.5	5	5.5	4.5	5	5.5	4.5	5	5.5	4.5	5	5.5	V
	74 Family	4.75	5	5.25	4.75	5	5.25	4.75	5	5.25	4.75	5	5.25	
High-level output current, I_{OH}				−800			−800			−800			−1000	μA
Low-level output current, I_{OL}				16			16			16			20	mA
Operating free-air temperature, T_A	54 Family	−55		125	−55		125	−55		125	−55		125	°C
	74 Family	0		70	0		70	0		70	0		70	

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

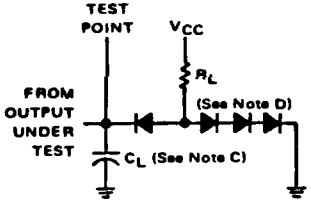
PARAMETER		TEST FIGURE	TEST CONDITIONS†		SERIES 54 / SERIES 74 '13			SERIES 54 / SERIES 74 '14			SERIES 54 / SERIES 74 '132			SERIES 54S / SERIES 74S 'S132			UNIT
					MIN	TYP‡	MAX	MIN	TYP‡	MAX	MIN	TYP‡	MAX	MIN	TYP‡	MAX	
V_{T+}	Positive-going threshold voltage	8	V_{CC} = 5 V		1.5	1.7	2	1.5	1.7	2	1.5	1.7	2	1.6	1.77	1.9	V
V_{T-}	Negative-going threshold voltage	9	V_{CC} = 5 V		0.6	0.9	1.1	0.6	0.9	1.1	0.6	0.9	1.1	1.1	1.22	1.4	V
	Hysteresis ($V_{T+} - V_{T-}$)	8, 9	V_{CC} = 5 V		0.4	0.8		0.4	0.8		0.4	0.8		0.2	0.55		V
V_I	Input clamp voltage	3	V_{CC} = MIN,	I_I = §			−1.5			−1.5			−1.5			−1.2	V
V_{OH}	High-level output voltage	9	V_{CC} = MIN, V_I = V_{T-} min, I_{OH} = MAX	54 Family	2.4	3.4		2.4	3.4		2.4	3.4		2.5	3.4		V
				74 Family	2.4	3.4		2.4	3.4		2.4	3.4		2.7	3.4		
V_{OL}	Low-level output voltage	8	V_{CC} = MIN, V_I = V_{T+} max, I_{OL} = MAX			0.2	0.4		0.2	0.4		0.2	0.4			0.5	V
I_{T+}	Input current at positive-going threshold	8	V_{CC} = 5 V,	V_I = V_{T+}		−0.65			−0.43			−0.43			−0.9		mA
I_{T-}	Input current at negative-going threshold	9	V_{CC} = 5 V,	V_I = V_{T-}		−0.85			−0.56			−0.56			−1.1		mA
I_I	Input current at maximum input voltage	4	V_{CC} = MAX,	V_I = 5.5 V			1			1			1			1	mA
I_{IH}	High-level input current	4	V_{CC} = MAX,	V_I = 2.4 V			40			40			40				μA
				V_I = 2.7 V												50	
I_{IL}	Low-level input current	5	V_{CC} = MAX,	V_{IL} = 0.4 V		−1	−1.6		−0.8	−1.2		−0.8	−1.2				mA
				V_{IL} = 0.5 V												−2	
I_{OS}	Short-circuit output current♦	6	V_{CC} = MAX		−18		−55	−18		−55	−18		−55	−40		−100	mA
I_{CC}	Supply current, Total, output high	7	V_{CC} = MAX			14	23		22	36		15	24		28	44	mA
	Supply current, Total, output low					20	32		39	60		26	40		44	68	
	Supply current, Average per gate		V_{CC} = 5 V,	50% duty cycle		8.5			5.1			5.1			9		

†For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions.
‡All typical values are at V_{CC} = 5 V, T_A = 25°C.
§I_I = −12 mA for SN54'/SN74' and −18 mA for 'S132.
♦Not more than one output should be shorted at a time, and for 'S132, duration of output short-circuit should not exceed one second.

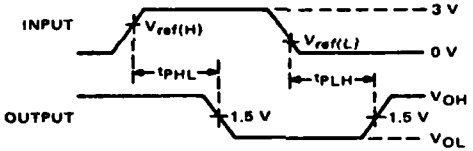
switching characteristics, V_{CC} = 5 V, T_A = 25°C

TYPE	TEST CONDITIONS	t_{PLH} (ns) Propagation delay time, low-to-high-level output		t_{PHL} (ns) Propagation delay time, high-to-low-level output	
		TYP	MAX	TYP	MAX
'13	C_L = 15 pF, R_L = 400 Ω	18	27	15	22
'14, '132		15	22	15	22
'S132	C_L = 15 pF, R_L = 280 Ω	7	10.5	8.5	13

PARAMETER MEASUREMENT INFORMATION



LOAD CIRCUIT



VOLTAGE WAVEFORMS

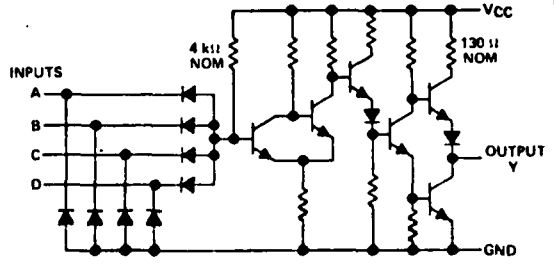
NOTES: A. The input waveform is supplied by a generator with the following characteristics: Z_{out} = 50 Ω and PRR ≤ 1 MHz. Rise and fall times between 10 and 90 percent points are 10 ns for SN54'/SN74' circuits and 2.5 ns for 'S132.

B. Reference voltages for SN54'/SN74' circuits are: $V_{ref(H)}$ = 1.7 V, $V_{ref(L)}$ = 0.9 V. Reference voltages for 'S132 are: $V_{ref(H)}$ = 1.8 V, $V_{ref(L)}$ = 1.2 V.

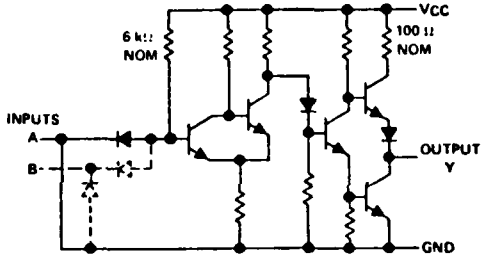
C. C_L includes probe and jig capacitance.

D. All diodes are 1N916 or 1N3064.

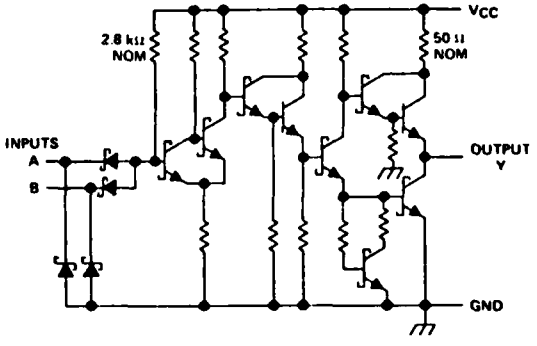
schematics (each gate)



'13 CIRCUITS



'14, '132 CIRCUITS



'S132 CIRCUITS

TYPICAL CHARACTERISTICS†

POSITIVE-GOING THRESHOLD VOLTAGE vs FREE-AIR TEMPERATURE

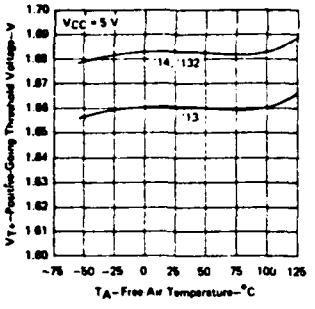


FIGURE 1

NEGATIVE-GOING THRESHOLD VOLTAGE vs FREE-AIR TEMPERATURE

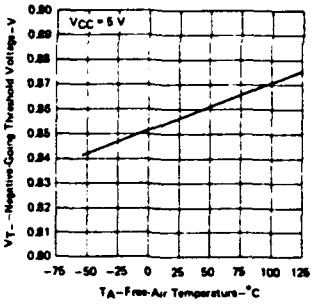


FIGURE 2

HYSTERESIS vs FREE-AIR TEMPERATURE

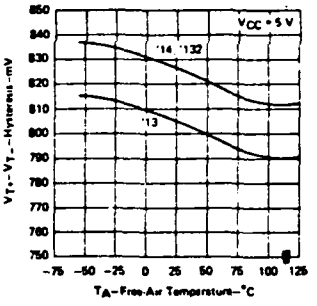


FIGURE 3

'13 DISTRIBUTION OF UNITS FOR HYSTERESIS

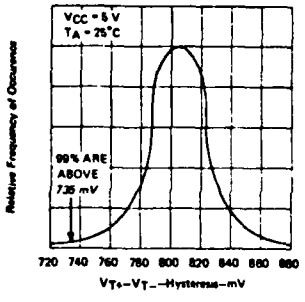


FIGURE 4

'14, '132 DISTRIBUTION OF UNITS FOR HYSTERESIS

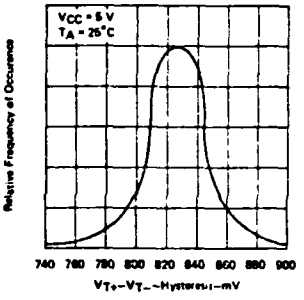


FIGURE 5

THRESHOLD VOLTAGES vs SUPPLY VOLTAGE

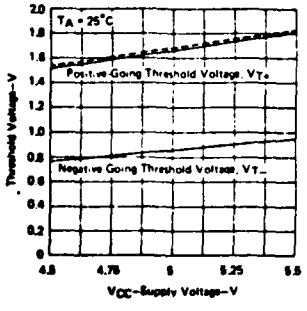


FIGURE 6

HYSTERESIS vs SUPPLY VOLTAGE

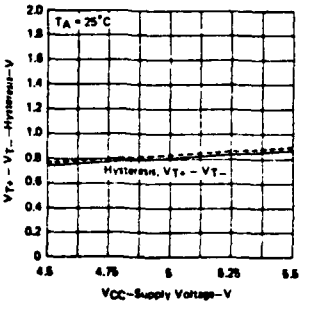


FIGURE 7

OUTPUT VOLTAGE vs INPUT VOLTAGE

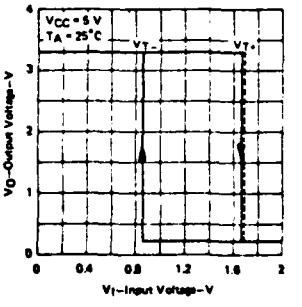
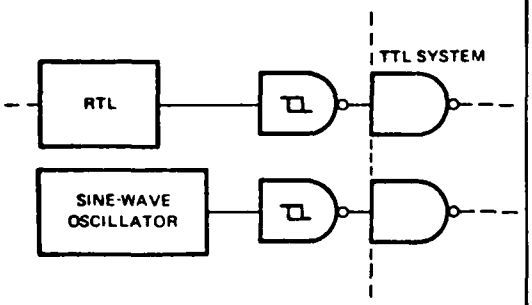


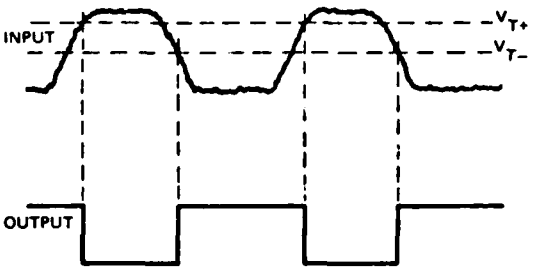
FIGURE 8

†Typical values of '13, '14, and '132 circuit types. Dashed lines in Figures 6, 7, and 8 are applicable for the '14 and '132 circuit types.

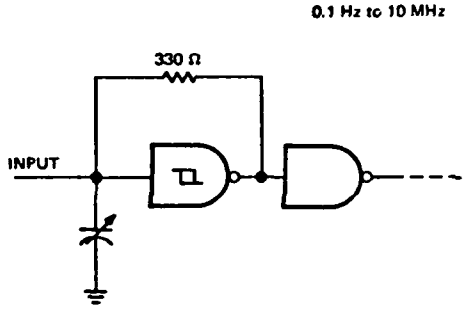
TYPICAL APPLICATION DATA



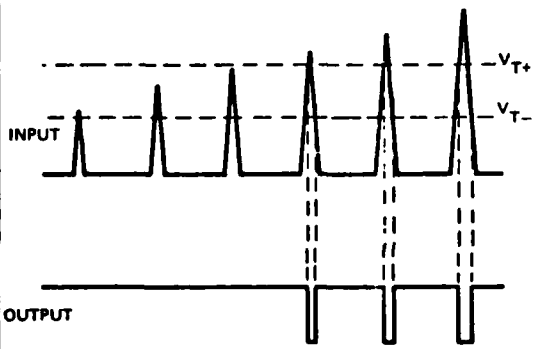
TTL SYSTEM INTERFACE
FOR SLOW INPUT WAVEFORMS



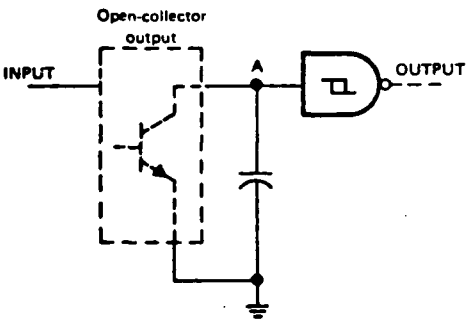
PULSE SHAPER

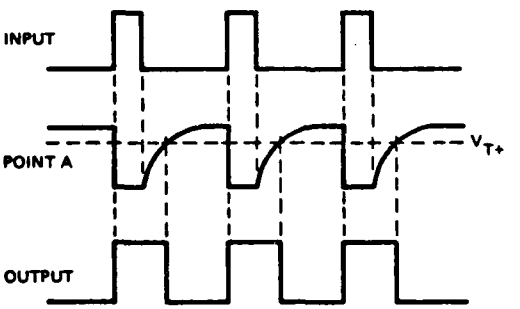


MULTIVIBRATOR



THRESHOLD DETECTOR





PULSE STRETCHER

Bibliography

1. Allen, G. Expansion of the Eclipse Digital Signal Processing System. MS Thesis. Wright-Patterson AFB, Ohio: School of Engineering, Air Force Institute of Technology, December,1982.

2. Birkmeier, W. P., et. al. "Pattern Recognition Techniques for the Detection of Epileptic Transients in EEG," IEEE Transactions on Biomedical Engineering BME-25: 213-217, May 1978.

3. Brenner, D. et.al. Evoked Magnetic Fields Reveal Different Visual Areas in Human Cortex. New York, N.Y.: New York University, June 1980.

4. Bronzino, J. D. "Utilization of Amplitude Histographs to Quantify the EEG Effects of Systemic Administration of Morphine in the Chronically Implanted Rat," IEEE Transactions on Biomedical Engineering BME-28: 673-677, October 1981.

5. Boxer, Aaron. "Designing For High Performance Data Acquisition," Computer Design, September 1983.

6. Cane, Dave and Steve Mullen. "Triple-Bus Architecture Gains Speed, Versatility," Computer Design, February 1984.

7. Cappellini, V., A. G. Constantinides, and P. Emiliani Digital Filters and their Applications. London: Academic Press, 1978.

8. Demott, D.M. "Cortical Micro-toposcopy," Medical Research Engineering: 23-29, October-December 1966.

9. ---- "Microtoposcopic Investigation of the Peristriate Cortex of the Squirrel Monkey," Medical Research Engineering, 10: 17-21, April-May 1971.

10. ---- Toposcopic Studies of Learning. Springfield: Charles C. Thomas, 1971.

11. De Weerd, J. P. C. "Facts and Fancies About a Posteriori 'Wiener' Filtering," IEEE Transactions on Biomedical Engineering BME-28: 252-257, March 1981.

12. Donchim, E. "Data Analysis Techniques in Average Evoked Potential Research," Average Evoked Potentials Methods, Results, and Evaluations. 199-236. San Francisco: National Aeronautics and Space Administration, September, 1969. (NASA SP-191).

13. Ervin, F.R. "On-Line Computer Techniques for Analysis of the Visual System," Symposium on the Analysis of Central

Nervous System and Cardiovascular Data Using Computer Methods: 35-51. Wasnington D.C.: National Aeronautics and Space Administration, October 1964.

14. Fitzgerald, G. H. The Development of a Two-Dimensional Multielectrode Array for Visual Perception Research in the Mammalian Brain. MS Thesis. Wright-Patterson AFB, Ohio: School of Engineering, Air Force Institute of Technology, December 1980. (AD A100763)

15. Gastaut, H. and H. Regis. "Visually-Evoked Potentials Recorded Transcranially in Man," Symposium on the Analysis of Central Nervous System and Cardiovascular Data Using Computer Methods. 7-34. Washington D.C.: National Aeronautics and Space Administration, October 1964.

16. German, G. W. A Cortically Implantable Multielectrode Array for Investigating the Mammalian Visual Cortex. MS Thesis. Wright-Patterson AFB, Ohio: School of Engineering, Air Force Institute of Technology, December 1981.

17. Harding, G. F. A. "EEG Techniques, with Particular Reference to the Analysis of Spontaneous Activity," Chapter IV of The EEG in the Periodic Psychoses. Unpublished PH. D. Thesis. Birmingham, England: University of Birmingham, 1968.

18. Hayes, R. E. The First Cortical Implant of a Semiconductor Multielectrode Array: Data Analysis. MS Thesis. Wright-Patterson AFB, Ohio: School of Engineering, Air Force Institute of Technology, December, 1983.

19. Hensley, R.W. and D.C.Denton. The First Cortical Implant of a Semiconductor Multielectrode Array: Electrode Development and Data Collection. MS Thesis. Wright-Patterson AFB, Ohio: School of Engineering, Air Force Institute of Technology, December 1982.

20. Hubel, D.H. and T.N. Wiesel. "Receptive Fields, Binocular Interaction and Functional Architecture in the Cat's Visual Cortex," Journal of Physiology 160: 406-454, 1962.

21. Jansen, B.H., J.R.Bourne and J.W. Ward. "Autoregressive Estimation of Short Segment Spectra for Computerized EEG Analysis," IEEE Transactions on Biomedical Engineering BME-28: 630-638, September 1981.

22. ---- "Spectral Decomposition of EEG Intervals Using Walsh and Fourier Transforms," IEEE Transactions on Biomedical Engineering BME-28: 836-838, December 1981.

23. Kaprisky, M.J. A Proposed Model for Visual Processing in the Human Brain. Urbana: University of Illinois Press, 1966.

24. Klein, F. F. and D. A. Davis. "The Use of Time Domain Analyzed EEG in Conjunction with Cardiovascular Parameters for Monitoring Anesthetic Levels," IEEE Transactions on Biomedical Engineering BME-28: 36-40, January 1981.

25. Lenz, Jr., J.E. and E.F. Kelly. "A Potential Difficulty in A/D Conversion Using Microcomputer Systems," IEEE Transactions on Biomedical Engineering BME-27: 668-669, November 1980.

26. Levreault, Jr., John E. "High-Speed Data Acquisition Design System," Electronic Products: 89-92, March 4, 1983.

27. Lim, A. J. and W. D. Winters. "A Practical Method for Automatic Real-Time EEG Sleep State Analysis," IEEE Transactions on Biomedical Engineering BME-27: 212-220, April 1980.

28. Lorente deNo', R. "Cerebral Cortex: Architecture, Intracortical Connections, Motor Projections," Physiology of the Nervous System. 3rd ed., by J.F. Fulton, Oxford: Oxford University Press, 1949.

29. Margerison, J. H., P. St. John-Loe, and C. D. Binnie. "Electroencephalography," in P. H. Venables and I. Martin, Eds. Manual of Psycho-physiological Methods. Amsterdam: North-Holland, 1967. As quoted by G. F. A. Harding. "EEG Techniques, with Particular Reference to the Analysis of Spontaneous Activity," Chapter IV of The EEG in the Periodic Psychoses. Unpublished Ph. D. Thesis. Birmingham, England: University of Birmingham, 1968.

30. Maybeck, P. S. Stochastic Models, Estimation, and Control Volume 1. New York: Academic Press, 1979.

31. Mountcastle, V. B. "Modality and Topographic Properties of Single Neurons of Cat's Somatic Sensory Cortex." Journal of Neurophysiology 20: 408-434, 1957.

32. Nunez, D.L. "A Study of Origins of the Time Dependencies of Scalp EEG: I-Theoretical Basis," IEEE Transactions on Biomedical Engineering BME-28: 271-280, March 1981.

33. ----- "A Study of the Origins of the Time Dependencies of Scalp EEG: II-Experimental Support of Theory," IEEE Transactions on Biomedical Engineering BME-28: 281-288, March 1981.

34. Oppenheim, A. V. and R. W. Schafer. Digital Signal Processing. Englewood Cliffs: Prentice-Hall, 1975.

35. Perry, N. W., Jr. and D. G. Childers. The Human Visual Evoked Response: Method and Theory. Springfield Illinois: Charles C. Thomas, 1969.

36. Rabiner, L. R., R. W. Schafer and D. Dlugos. "Correlation

Method for Power Spectrum Estimation" in C. J. Weinstein, et. al., Eds. Programs for Digital Signal Processing. New York: IEEE Press, 1979.

37. Rader, C. M. "An Improved Algorithm for High Speed Autocorrelation with Applications to Spectral Estimation," IEEE Transactions on Audio and Electroacoustics AU-18: 439-441, December, 1970.

38. Remond, A. "Topological Aspects of the Organization, Processing and Presentation of Data," Symposium on the Analysis of Central Nervous System and Cardiovascular Data Using Computer Methods. 73-93. Washington, D.C.: National Aeronautics and Space Administration, October, 1964. (NASA SP-72).

39. Shereff, Jess. "Designing OEM System," Electronic Design 101-114, July 26, 1984.

40. Sjontoft, E. "A Deconvolution Procedure for Use in Extracting Features in Average Evoked Response EEG Signals," IEEE Transactions on Biomedical Engineering BME-27: 227-230, April 1980.

41. Smith, S. D. "Walsh Versus Fourier Estimators of the EEG Power Spectrum," IEEE Transactions on Biomedical Engineering BME-28: 790-793, November 1981.

42. Tatman, J. A. A Two-Dimensional Multielectrode Microprobe for the Visual System. MS Thesis. Wright-Patterson AFB, Ohio: School of Engineering, Air Force Institute of Technology December 1979. (AD A080378).

43. Quick Choice Users Manual, Masscomp Corporation, 1983.

VITA

John David Collier was born the son of John F. and Margaret A. Collier on 25 August 1947 in Camden, New Jersey. He graduated from High School in Mayfield, New York in June 1965. He received an Associates in Science Degree in Electrical Technology from Fulton-Montgomery Community College, Johnstown, New York in May 1967. After attending Pratt Institute in Brooklyn, New York he enlisted in the Air Force in August 1968. Upon completion of technical training at Lowry AFB, Colorado he was assigned to Hahn AB, Germany. He was accepted as a candidate for a bachelors Degree in Electrical Engineering in the Airmen's Education and Commissioning Program in January 1973 at the University of Missouri-Columbia, Columbia, Missouri. After graduation in May 1975 he attended Officer Training School and was commissioned in August 1975. Upon completion of Communications-Electronics Officer's course at Keesler AFB, Mississippi he was assigned to Rhein-Main AB, Germany as a NAVAIDS Evaluation Officer. Subsequent assignments include Scott AFB, Illinois and RAF Uxbridge, United Kingdom. He entered the School of Engineering, Air Force Institute of Technology in June 1983. He is a member of Eta Kappa Nu and Tau Beta Pi engineering honor societies.

Permanent Address: 167 East State Street
Gloversville, New York 12078

VITA

Deanne Barnett Mahoney was born the daughter of Beverly James and Margie Miller Barnett on 4 March 1958 at Itazuke A.F.B., Japan. She graduated from Mascoutah Community High School, Mascoutah, Illinois in 1976. She entered the United States Air Force Academy in June of 1976 and graduated in May of 1980 with a Bachelor of Science in Engineering Physics. Commisioned a 2nd Lieutenant upon graduation, her initial assignment was to Cheyenne Mountain Complex, Colorado Springs, Colorado where she served as an Orbital Analyst. She entered the School of Engineering, Air Force Institute of Technology in June 1983.

Permanent Address: 18884 Chaparral Drive
Penn Valley, California
95946

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b RESTRICTIVE MARKINGS
2a SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution unlimited.
2b DECLASSIFICATION/DOWNGRADING SCHEDULE	
4 PERFORMING ORGANIZATION REPORT NUMBER(S) AFIT/GE/GSO/ENG/84D-20	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a NAME OF PERFORMING ORGANIZATION School of engineering 6b. OFFICE SYMBOL (If applicable) AFIT/ENG	7a. NAME OF MONITORING ORGANIZATION
6c ADDRESS (City, State and ZIP Code) Air Force Institute of Technology Wright-Patterson AFB, Ohio 45433	7b. ADDRESS (City, State and ZIP Code)
8a NAME OF FUNDING/SPONSORING ORGANIZATION 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c ADDRESS (City, State and ZIP Code)	10 SOURCE OF FUNDING NOS PROGRAM ELEMENT NO / PROJECT NO / TASK NO / WORK UNIT NO
11 TITLE (Include Security Classification) See Box 19	
12 PERSONAL AUTHOR(S) John D. Collier, B.S., Capt, USAF and Deanne B. Mahoney, B.S., Capt, USAF	
13a TYPE OF REPORT MS Thesis 13b TIME COVERED FROM ____ TO ____	14 DATE OF REPORT (Yr, Mo, Day) 1984 December 15 PAGE COUNT 216
16 SUPPLEMENTARY NOTATION	

17 COSATI CODES		
FIELD	GROUP	SUB GR
06	04	
06	05	

18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
Bionics
Electroencephalography (EEG)
Visual Evoked Response (VER)

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

Title: THE FIRST CORTICAL IMPLANT OF A SEMICONDUCTOR MULTIELECTRODE ARRAY: CONTINUED DATA ANALYSIS

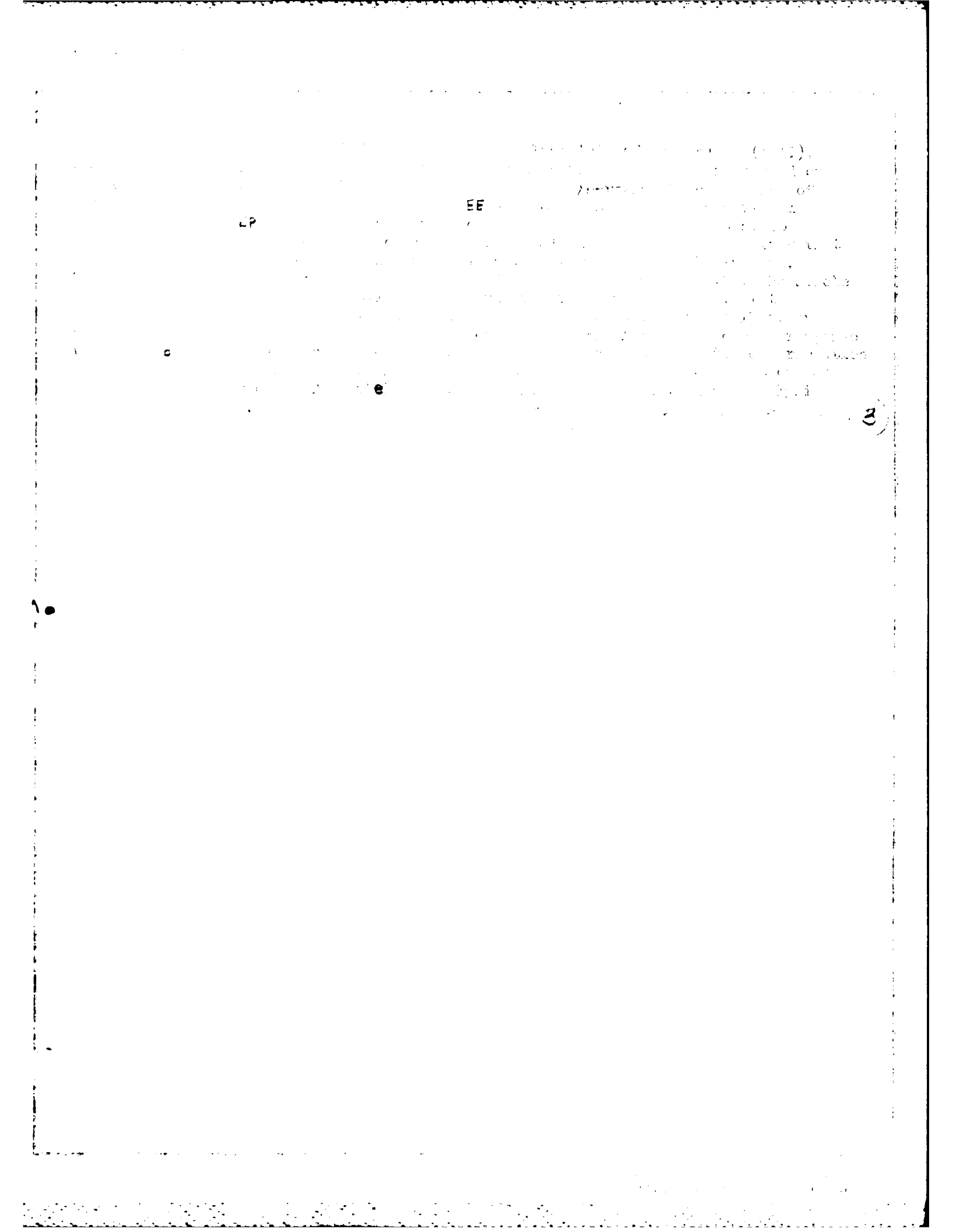
Thesis Chairman: Matthew J. Kabrisky, PhD
Professor, Department of Electrical Engineering

Approved for Public Release: IAW AFR 190-17
LYNN E. WOLAVER
Dean for Research and Professional Development
Air Force Institute of Technology (AU)
Wright-Patterson AFB OH 45433

20 DISTRIBUTION/AVAILABILITY OF ABSTRACT UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT ☐ DTIC USERS ☐	21 ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a NAME OF RESPONSIBLE INDIVIDUAL Matthew J. Kabrisky, PhD	22b TELEPHONE NUMBER (Include Area Code) 513-255-5276 22c OFFICE SYMBOL AFIT/ENG

DD FORM 1473, 83 APR EDITION OF 1 JAN 73 IS OBSOLETE

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE


EE
LP

END

FILMED

5-85

DTIC